Development of Effective & Accessible Approach to Analyze Unsteady State Plant Performance

By

Muralidhar Satuluri

Submitted to the graduate degree program in Chemical and Petroleum Engineering and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

_____

Chairperson: Dr. Colin S. Howat

_____

Dr. Kyle Camarda

_____

Dr. Don Green

_____

Dr. Aaron Scurto

_____

Dr. Karan Surana

April 22, 2011

The Dissertation Committee for Muralidhar Satuluri

certifies that this is the approved version of the following thesis:

Development of Effective & Accessible Approach to Analyze Unsteady State Plant Performance

_____

Chairperson: Dr. Colin S. Howat

April 22, 2011

Acknowledgments

I would like to first thank my committee members. They have been very kind in agreeing to the defense at short notice and have provided valuable thoughts on my work.

Many friends helped me during my PhD. Piyush, Prakash, Milind & Aravind have given me much company. Rajesh, who showed me how fragile life is, was a constant source of support. My colleagues at KUCIC have provided the most fun filled year. I will not forget the shows and all that went into them.

Dave, Luke & Pat, you guys were too much fun. I had to follow you to Houston. Elim, you were my only KTL colleague for the later half of my PhD. Thanks for all your help and wish you the best moving forward.

Dr. Howat: He has been my well wisher, mentor and best friend. Ten years ago when I joined KTL, I didn't know that it was the best decision of my life. I can't express how thankful I am to my advisor. There is an ancient Indian hymn that equates the teacher to the Gods. When I think of that hymn, I think of Dr. Howat.

Last, but not the least, I have to thank Hema. This dissertation would have been just a dream without her.

Abstract

Measurements from chemical processes contain error. They are reconciled with the process model to improve their accuracy. A method to reconcile unsteady state, i.e. time dependent, plant data using process simulators and dedicated optimization software is presented. The reconciliation is carried out in the optimization software and the process model is supplied in the form of dynamic process simulations built in the simulator. The optimization software and the process simulator are interconnected using OLE (Object Linked Embedding) technology from Microsoft.

Mathematical models representing dynamic process operation are in the form of Differential Algebraic Equations (DAE). Existing methods require the process engineer to program the reconciliation routine and a solution to the DAE model using advanced numerical techniques. This poses high entry barrier. Proposed method instead relies on rating based process simulation which is familiar to a typical process engineer. Many process plants maintain simulation models further reducing the effort needed to develop reconciliation program. The only additional requirements for using this method are working knowledge of mathematical optimization software and OLE technology.

The Rating Based Reconciliation (RBR) method developed in this work is tested in three case studies of increasing complexity. Error in several of the measurements reduced after reconciliation. Accuracy of the results from the proposed method is comparable to those from literature. The accuracy improves by using wavelet denoising prior to reconciliation.

Table of Contents

Table of Figures

Table of Figures

Nomenclature

Symbols	Description
a	Coefficients of the trial solution in Method of Weighted Residuals
f	Process model containing set of mathematical equations
n_c	Number of collocation points
n_m	Number of measurements at any given time
n_p	Number of parameters at any given time
n_u	Number of input variables.
n_x	Number of state variables.
n_y	Number of output variables.
R	Variance Covariance matrix
Rd	Residual in the Method of Weighted Residuals
u	Vector of input variables.
x	Vector of state variables
y	Vector of output variables
Z	Vector of measurements
$\hat{Z}$	Vector of reconciled measurements
$\hat{\theta}$	Vector of unmeasured variables.
σ_i	Standard deviation of ith measurement.
σ^u	Standard deviation in the input variables.
σ^x	Standard deviation in the state variables.
σ^y	Standard deviation in the output variables.

1. Introduction

Plant measurements are corrupted by random and systematic errors. The errors are introduced by the sensors and fluctuations inherent in the process. Consequently, constraints such as material and energy balances are not met. As a result any description of the process using a model built from the measurements isn't correct. Unmeasured variables such as stage temperatures and heat transfer coefficients calculated from the measurements will be uncertain. The goal of this dissertation is to build a framework that can perform the following functions:

1. Estimate values for measurements that close the constraints.
2. Reduce uncertainty in the measurements.
3. Estimate unmeasured process variables from the improved measurement estimates.

Measurements and parameters that are satisfy the constraints and with lower uncertainty provide a better picture of the plant operation. They result in tangible benefits such as:

1. Accurate product flow rate data is valuable in calculating the process plant economics.
2. Accurate process parameters such as heat transfer coefficient and column efficiency can be used to monitor the performance of the equipment. Any deterioration in the equipment performance will be detected in changing values for these parameters.

3. If reconciled measurements are significantly different from raw measurements, they imply a problem with measurement sensor.

1.1 Motivation from Industrial Projects

Two industrial projects provided motivation for this work. The first project involved building Operator Training Software for the butadiene recovery section in a SBR (Styrene Butadiene Polymer) polymerization plant. The process in question (Figure 1-1) contained a sequence of flash vessels, heat exchangers, a stripper and an absorber to remove butadiene from air before the latter is vented. The second project analyzed and estimated the pressure drop across a 167 mile pipeline (Figure 1-2). This pipeline, along with the pressure boosting stations in between, is used to transport crude oil from salt domes to storage tanks in a far away refinery.

Figure 1-1. Butadiene Recovery Area in SBR Polymerization Plant.

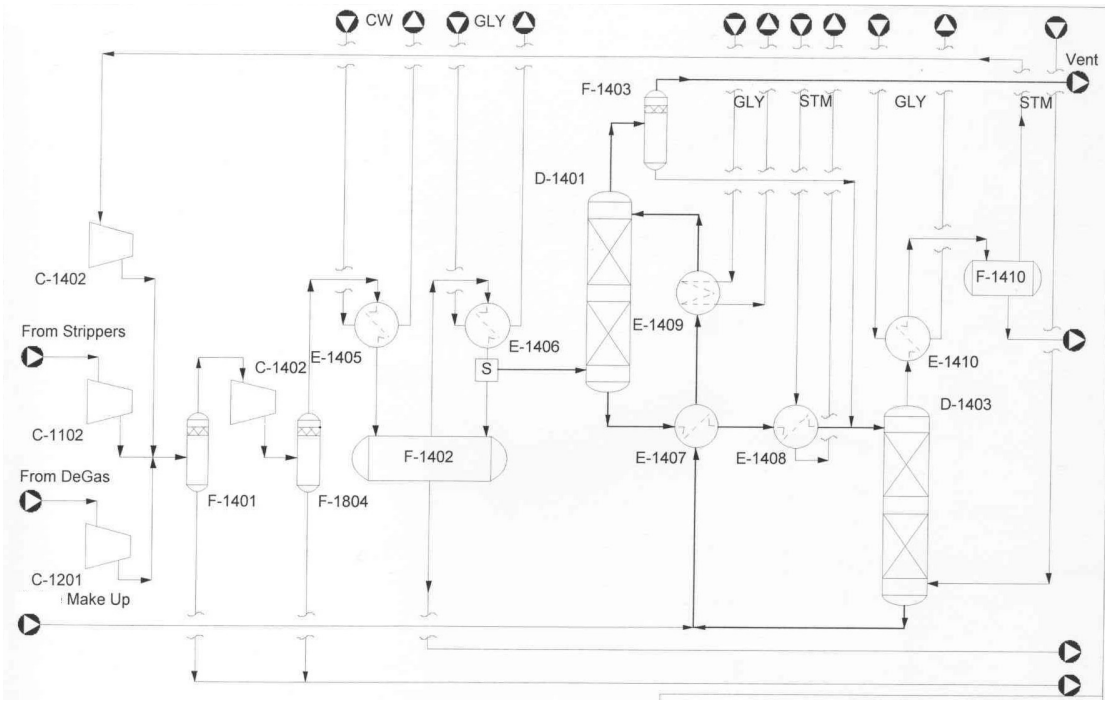
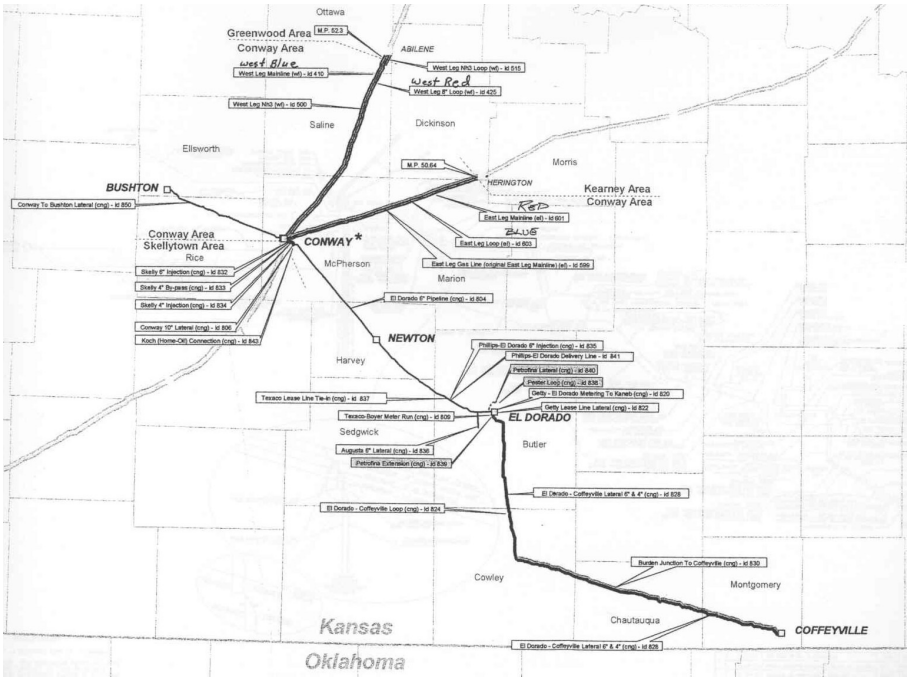


Figure 1-2. Oil Pipeline Drawing for the Pressure Drop Analysis Project



The first step in both projects was to build a preliminary model of the operation. These models were based on fundamental chemical engineering principles. In both cases, however variables with unknown values were part of the models (e.g. mass transfer coefficients, pump curve coefficients etc). Values for such unknown variables must be estimated so that the model predictions match the measurements from the field. This estimation is problematic because:

1. Steady state models were used while the projects involved unsteady state operation. The unsteady nature in the SBR plant was due to the significant holdup in the operation. That in the pipeline project was due to the varying destination

pressure, i.e. pressure at the bottom of the storage tank changes as the tank gets filled.

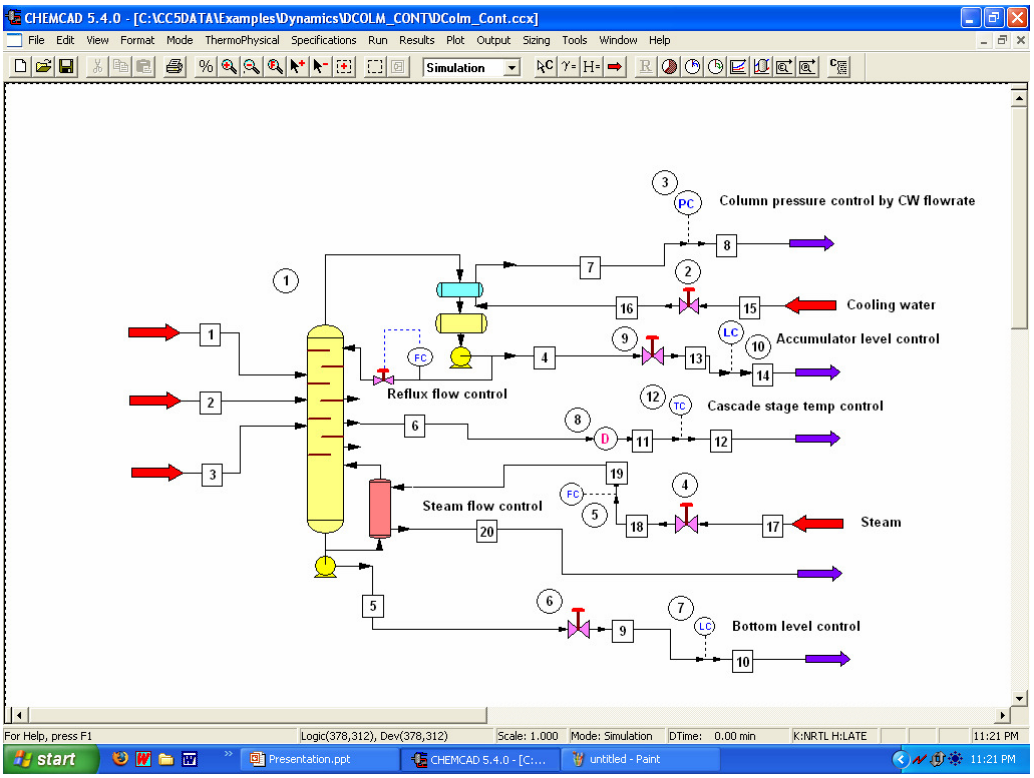
2. Measurements contained error. Consequently, they do not satisfy physical laws used in the model. Before estimating unknown variables the measurements must be adjusted to ensure they satisfy the model.

A systematic approach is needed for reconciling plant measurements with the corresponding process model and to estimate any unknown variables. Kuehn and Davidson (1961) laid the foundations of a methodology, called Data Reconciliation, addressing this problem. Data Reconciliation is a process of making least possible adjustments to the measurements such that the model equations are satisfied. The adjustment to a measurement will be proportional to its standard deviation. If one measurement (e.g. pressure) has higher standard deviation, then it will be adjusted more than the others.

Satuluri (2003) implemented the data reconciliation method for analyzing phase equilibrium data measured at Kurata Thermodynamics Laboratory, KTL (Cheng, 2003; Canaday, 2003). The phase equilibrium data consisted of pressure, temperature, volume and bulk composition measurements. The model contained mass, energy balance and phase equilibrium relations. Satuluri built a reconciliation program that would minimally adjust the lab measurements in order to satisfy the model equations and estimate the infinite dilution activity coefficient (γ_1^{∞}). This reconciliation method was limited to steady state situations such as the laboratory methods.

There was a need to extend the reconciliation method to unsteady state situations. Several researchers proposed reconciliation methods that can be applied to unsteady state processes (Edgar and Liebman, 1991 and Vachhani *et al*, 2005). They all share one feature: extensive programming. To apply these methods to a single unit operation such as a flash vessel or a reactor, the plant engineer has to derive the dynamic model equations and program their solution using advanced numerical methods such as orthogonal collocation and nonlinear optimization. Process plants in reality consist of a web of multiple units with process controls and recycle streams (Figure 1-3). Examples are the SBR plant and the crude oil pipeline projects mentioned earlier. Applying existing reconciliation methods to such situations would take Herculean effort.

Figure 1-3. Distillation Column With Controls (Chemcad Documentation)



This work introduces a reconciliation framework that significantly reduces the programming effort through the use of process simulators and Microsoft Excel. Building a reconciliation program using these two tools mainly consists of two steps

1. Build a dynamic simulation of the process in the simulator of choice (e.g. Aspen Dynamics, Hysys and CHEMCAD).
2. Develop Excel control module, i.e. a VBA program in Microsoft Excel that can provide input to the simulation, run and access the results from the simulation.

Many plant engineers use process simulators and Excel as their tools of trade. Plants also maintain simulations representing their process, albeit in steady state mode. Process simulators (e.g. Hysys and Aspen Dynamics) allow these models to be extended to dynamic mode easily. The dynamic sequential-modular representation of the process is developed in the process simulator. The reconciliation routine is developed in MATLAB or Excel. The Excel control module is developed to interact with the simulator, providing input, controlling the run and extracting the necessary output. The Excel control module is developed to also interact with the reconciliation routine such that simulation and measured information is fed to the routine and the reconciled measurements are returned. The Excel module then feeds the new measurements to the simulator and the process is repeated until the constraints are met with minimal adjustment to the measurements. This skeletal description provides the overview of the primary emphasis of this work.

An additional aspect treats the data independent of the underlying chemical engineering fundamentals. The dynamic nature of the measurements, i.e. changes with time, must be captured properly in the reconciled measurements. If not, then the resultant reconciled measurements do not properly reflect the true behavior and any unmeasured variable estimates based thereon will be uncertain. Wavelet denoising (Jansen, 2001) is used in conjunction with the above to filter random behavior from the underlying dynamic behavior. Therefore, the Excel control module is developed to interact with a denoising tool.

The results of this work show that the extensive programming required in the literature discussions is eliminated through use of commercially available process simulation tools, the extensive programming required for reconciliation optimization is eliminated through use of commercially available software packages and the dynamic characteristics are properly captured using measurement pre-treatment minimizing the corruption of the plant performance analysis.

2. Literature Review

2.1 Background

The two sections of the literature discuss the mathematical foundation for reconciliation and the solution methods to solve reconciliation problems.

2.1.1 Steady State Reconciliation

Reconciliation is the process of calculating measurement estimates ($\hat{Z}$) that satisfy the process constraints (i.e. physical and chemical laws) and stay close to the measurements (Z) in value. During this calculation any unmeasured variables ($\hat{\theta}$, e.g. tray efficiency, composition) are also estimated. Mathematically this is equivalent to minimizing the square of the weighted difference between the measurements and measurement estimates while satisfying the process constraints ($f(\hat{Z},\hat{\theta})$). The weights in the objective function are the standard deviations in the measurements (σ_i). The objective function is derived from assuming that the measurements only contain normally distributed random error (Knepper and Gorman, 1980).

$$\min_{\hat{Z}_i,\hat{\theta}_j} \quad \sum_{i=1}^{n_m}\left(\frac{Z_i-\hat{Z}_i}{\sigma_i}\right)^2 \qquad \text{(II-1)}$$

Such that $\quad f(\hat{Z},\hat{\theta})=0 \qquad \text{(II-2)}$

Where n_m [=] number of measurements,

j [=] *1* to n_p, n_p is the number of unmeasured variables.

Number of measurements, constraints and unmeasured variables are related as follows (Knepper and Gorman, 1980): n_m > number of constraints > n_p

The problem of data reconciliation was first discussed by Kuehn and Davidson (1961) in the context of computer based control. Britt and Luecke (1973) and Knepper and Gorman (1980) developed the statistical and probabilistic underpinnings of reconciliation. They also provided the algorithms for solving it. MacDonald and Howat (1986) applied these algorithms to computer generated measurements of a non-equilibrium flash operation. They proved that the measurement estimates better represented the true behavior and expanded the methodology to simultaneously estimate the process variable, flash efficiency.

2.1.2 Unsteady State Reconciliation

Real world processes do not operate in steady state. The data reconciliation problem for unsteady state process is given by equations (II-3) to (II-5). All the measurements and the unmeasured variables are functions of time. The constraints consist of ordinary differential equations (e.g. mass and enthalpy balances) and algebraic equations (e.g. vapor liquid equilibrium). The differential term in the equations accounts for the changes in the process conditions with time.

$$\min_{\hat{Z}_i(t),\hat{\theta}_j(t)} \quad \sum_{i=1}^{n_m}\left(\frac{Z_i(t)-\hat{Z}_i(t)}{\sigma_i(t)}\right)^2 \tag{II-3}$$

Such that $$f\left(\frac{d\hat{Z}(t)}{dt},\hat{Z}(t),\hat{\theta}(t)\right)=0 \tag{II-4}$$

$$g(\hat{Z}(t),\hat{\theta}(t))=0 \tag{II-5}$$

Many solutions to the above problem (including this work) use an approach called Moving Horizon Approach.

2.1.3 Moving Horizon Approach

Measurements are related to one another across time. For example, a disturbance in the feed flow to a column affects the distillate composition few minutes later. The delay is through the time required for the disturbance to propagate through the column holdup. In this fashion, current measurements are related to those from the past through the holdup. Therefore, data from the past must also be used to reconcile the current measurements. Past data must also be used to solve differential equations. The entire data set back to time zero need not be used because: the current process conditions are most affected by the time steps closest to these measurements; and, all previous reconciled estimates incorporate the initial conditions, albeit to an ever lesser degree.

The number of measurements from the past which are used for reconciling the current measurements is called the history horizon. For example, if measurements are sampled every minute then a history horizon of 5 minutes would mean that to reconcile the current measurements, the last five measurement sets are also included in the problem. The length of the history horizon always stays constant. Figure 2-1 & Figure 2-2 show how the past measurements are selected for reconciling the current conditions. At $t = 6$ minutes, measurements from 2^{nd} min to 6^{th} minute are reconciled together. Only the reconciliation result for the 6^{th} minute is stored. For reconciling next set of measurements that arrive at the 7^{th} minute, data from 3^{rd} minute onwards is used. Here the

reconciliation results for only the 7th minute are stored. This process repeats. This approach is called the moving window estimation. It is used in this work and also by all the unsteady state reconciliation methods in the literature. Jang *et al*. (1986) and Kim *et al*. (1991) first used this approach for solving optimization problems involving differential algebraic equations.

The unsteady state reconciliation problem is rephrased in Equations (II-6) to (II-8) using the moving window approach. The objective function and the constraints consist of measurements only from the history horizon, i.e. the past *h* measurement sets.

$$\min_{\hat{Z}_{i,k},\hat{\theta}_{j,k}} \quad \sum_{i=1}^{n_m}\sum_{k=1}^{h+1}\left(\frac{Z_{i,k}-\hat{Z}_{i,k}}{\sigma_{i,k}}\right)^2 \text{, where } h=\frac{t_{end}-t_{ini}}{\Delta t} \qquad \text{(II-6)}$$

$$\text{Such that} \qquad f\left(\frac{d\hat{Z}}{dt},\hat{Z},\hat{\theta}\right)\Big|_{t_{ini}+(k-1)\Delta t}=0\text{, where } k = 1 \text{ to } (h+1) \qquad \text{(II-7)}$$

$$g(\hat{Z},\hat{\theta})\Big|_{t_{ini}+(k-1)\Delta t}=0 \qquad \text{(II-8)}$$

Figure 2-1. History Horizon @ t = 6 min

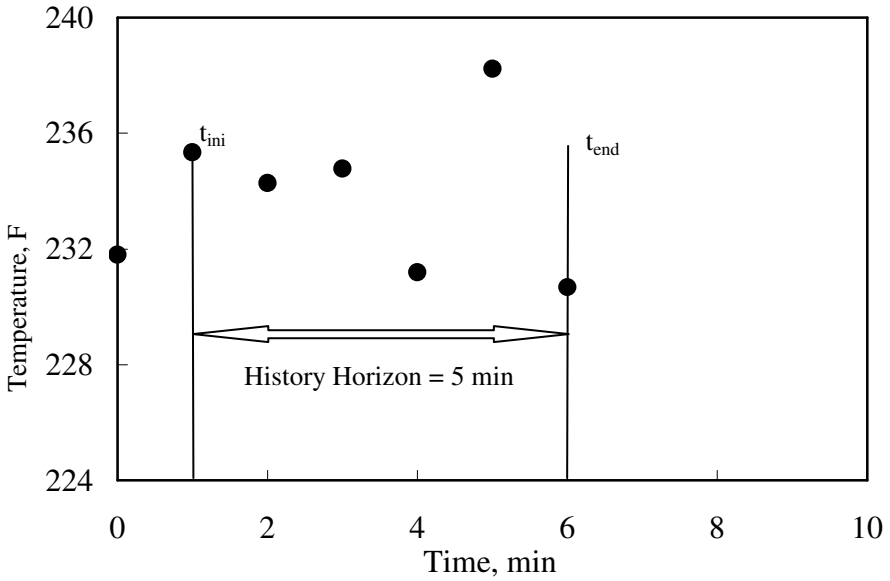
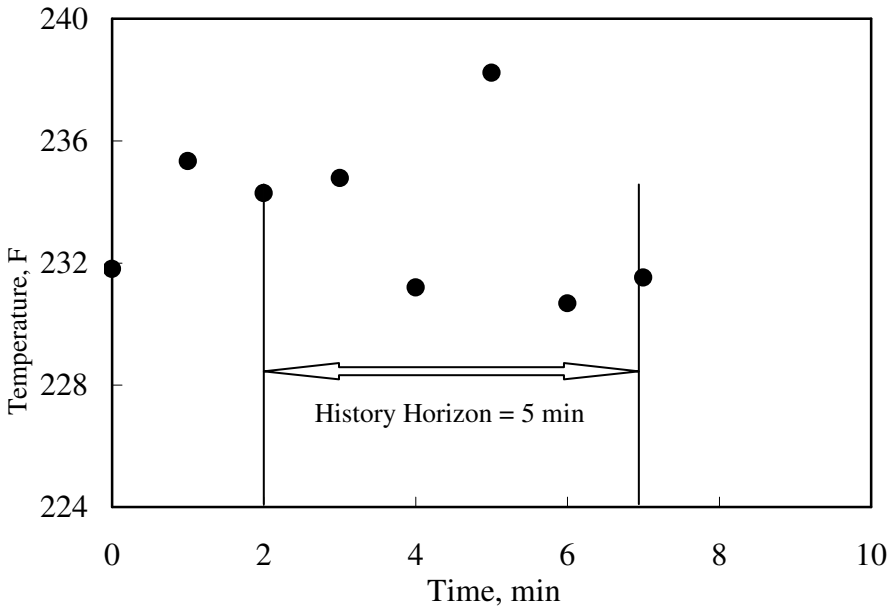


Figure 2-2. History Horizon @ t = 7 min.



2.2 Literature Methods

Literature cites three classes of methods (Prata *et al*, 2009) for solving the data reconciliation problem: Kalman filter based methods, methods using constrained nonlinear programming (e.g. Albuquerque and Biegler, 1995) and methods using model identification tools (e.g. Neural Networks used by Karjala and Himmelblau, 1994).

2.2.1 Kalman Filter Based Methods

The problem of improving the accuracy of dynamic process data was first addressed by Kalman (1960). His work was in the field of control theory. Kalman developed a recursive algorithm that adjusts the measurements according to the model. The algorithm consists of two steps

1. Prediction: Based on the estimate of the measurements at the current time, the measurements at the next time step are predicted. This is carried out by applying Euler's formula, $z_{k+1} = z_k + \Delta t \; f(t_k, z_k)$ where $\frac{dz}{dt} = f(t, z)$.
2. Correction: At the next time step the new measurements become available. These are combined with the predictions to arrive at new estimates of the current measurements. The combination is carried out optimally such that the precision in the resulting estimate is equal to or better than the prediction or the raw measurement.

Kalman filter offers one main advantage: speed. At each time step, the algorithm only requires evaluating algebraic equations. As a result the computations take very little time. Many researchers have applied Kalman filter to chemical processes. Venkateswarlu and Avantika (2001), Yildiz *et al* (2005) applied Kalman filter to batch distillation. Xie and Rohani (2001) used it in the case of a batch crystallizer.

While Kalman filter has been used by several researchers, it has significant disadvantages:

1. Kalman filter can reduce error only in state variables (Haseltine and Rawlings, 2005). State variables are the variables present in the differential term ($\frac{d(\ )}{dt}$) in the differential equations. Examples of state variables are the reactant concentrations and temperature in a continuous reactor, tray compositions and temperatures in a distillation column etc.
2. Traditional formulation assumes that the all the variables other than the State Variables are free of error (Haseltine and Rawlings, 2005).
3. The process model should not contain algebraic equations. The researchers who have used Kalman filter (Venkateswarlu and Avantika, 2001; Yildiz *et al*, 2005 & Xie and Rohani, 2001)) had to make assumptions in order to incorporate algebraic constraints into the differential equations. Some of the assumptions compromise the accuracy of the model (e.g. using Raoult's law for vapor liquid equilibrium).

The original Kalman filter was designed to handle only systems with linear differential equations. Extended Kalman filter was later developed to deal with nonlinear systems as well (Kopp and Oxford, 1963). In the extended filter, the nonlinear equations are linearized using first order Taylor series approximation. In the case of highly non-linear systems, first order approximations are not sufficient (Liebman *et al*., 1992). Haseltine and Rawlings (2005) and Jang et al (1987) report chemical engineering examples where extended Kalman filter fails to handle nonlinearities during reconciliation. Methods based on nonlinear programming (i.e. nonlinear optimization) are recommended instead.

2.2.2 Nonlinear Programming (i.e. Optimization) Based Methods

The early methods using nonlinear programming (Jang *et al*., 1987, Kim *et al*., 1991 and Ramamurthi *et al.*, 1993) to solve the dynamic reconciliation method were based on moving horizon approach and ODE integration. The initial conditions are the optimization variables. During each iteration, the measurement estimates at the beginning of the history horizon ($\hat{Z}_{i,1}$) are set. The process equations are then integrated to arrive at the measurement estimates for the reminder of the history horizon. By selecting the optimum initial conditions, the objective function is minimized. Since the model integration and optimization are performed in a sequence, it is called sequential approach.

$$\min_{\hat{Z}_{i,1}} \quad \sum_{i=1}^{n_m} \sum_{k=1}^{h+1} \left(\frac{Z_{i,k} - \hat{Z}_{i,k}}{\sigma_{i,k}} \right)^2 \text{, where } h = \frac{t_{end} - t_{ini}}{\Delta t} \qquad \text{(II-9)}$$

Such that $$f\left(\frac{d\hat{Z}}{dt}, \hat{Z}\right)\Big|_{t_{ini}+(k-1)\Delta t} = 0, \text{ where } k = 1 \text{ to } h+1 \qquad \text{(II-10)}$$

Liebman *et al*. (1991) present a reconciliation method that uses method of weighted residuals and Gaussian quadrature (Holland and Liapis, 1983; see section 8.1 and 8.2) to convert the differential equations in the model to algebraic equations. In this method, Lagrange interpolating polynomials are used to approximate the solution to the differential equation. The weights in the interpolating polynomials are chosen such that the integral of the error from approximation is zero in between any two measurements. This condition translates into residual equations (via Gaussian Quadrature) that must disappear at the collocation points, i.e. roots of the associated orthogonal polynomials (see Appendices 8.1 and 8.2). The residual equations are in the form of algebraic equations. This conversion transforms the dynamic optimization problem into a constrained optimization problem which can be solved using conventional optimization algorithms. Since the model equations and the optimization are solved simultaneously, this way of solution is called simultaneous approach. Biegler (1984) first used this approach for solving optimization problems involving differential algebraic constraints.

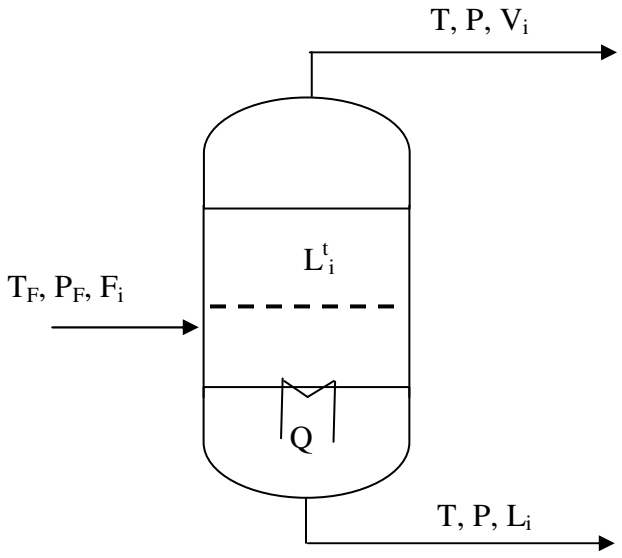
Kim *et al*. (1991) report that the sequential approach is faster than the simultaneous approach for the problem they choose. However, Vachhani *et al*.(2005), Liebman et al. (1991) and Robertson *et al*. (1996) state that in general the simultaneous approach is faster because sequential approach requires that the nonlinear differential equations be integrated during every iteration. Additionally Liebman *et al*. (1992) observe that sequential approach cannot handle algebraic constraints in the process model. This severely limits its applicability since most of the process equations contain phase equilibrium relations which are algebraic in nature.

Vachhani *et al*. (2005) developed a reconciliation method by converting the extended Kalman filter algorithm into a nonlinear programming problem with differential and algebraic equations as constraints. This problem is further converted into a constrained optimization problem through the method of weighted residuals and orthogonal collocation. The idea is to bestow Kalman filter with the ability to handle algebraic constraints while retaining its computational efficiency. Vachhani *et al*. (2005) report that the accuracy of reconciliation results from their work is equal to that from simultaneous approach proposed by Liebman *et al*. (1992). Furthermore, they report that the computational efficiency of their Kalman filter based method is an order of magnitude higher than that of the simultaneous approach (Liebman *et al*., 1992).

The simultaneous approach used for solving the reconciliation problem results in extremely large optimization problems (Alici and Edgar, 2002 and Albuquerque and Biegler, 1995). Consider the dynamic flash tank process shown in Figure 2-3. The process contains three components. Vapor holdup is ignored in the tank. The length of the moving window is 4 minutes (where 1 minute is the sampling interval) and the number of collocation points is 4. A rigorous formulation of the reconciliation problem using simultaneous approach leads to an optimization problem containing 327 optimization variables and 210 constraints. The optimization includes measurements and constraints at the collocation points that are in between the sampling instants. Their presence increases the size of the optimization. The Jacobian matrix for the resulting problem is large and sparse, but with predictable structure. Literature (Liebman *et al.*, 1992) exploits this predictability and uses optimization strategies that are well suited for large sparse Nonlinear Programming Problems.

The processes faced in the industries are substantially more complex than shown in Figure 2-3. They will very likely lead to nonlinear problems that have tens of thousands optimization variables. Programming the mathematics of solving such large optimization problems requires high level of skill in optimization along with chemical engineering. This dissuades process engineers from developing reconciliation routines for their processes.

Figure 2-3. Flash Tank



Albuquerque and Biegler (1995) solve the dynamic reconciliation problem by converting the differential equations into algebraic equations using implicit Runge Kutta methods. They apply the discretization at each measurement sample point. This minimizes the size of the optimization problem. The resulting constrained optimization problem is solved using SQP (Sequential Quadratic Programming) methods. However, their method still requires substantial programming effort in order to be used. Albuquerque and Biegler (1996) also propose variations of the least square objective function, used in reconciliation, in order to handle measurements containing gross error. Chen and Romagnoli (1998) use cluster analysis to identify measurements with gross error.

To address the drawback of optimization problem size, other studies proposed reconciliation using model identification tools such as neural networks and time series methods. These are discussed next.

2.2.3 Methods Using Model Identification Tools

Karjala and Himmelblau (1994) build a neural network model that can predict process conditions at time *t* using those at time *t-1*. The neural network model is trained offline using simulation data. The neural network is later used as the process model in solving the data reconciliation problem. They also use it for identifying measurements with gross error. This method requires large amount of off line data to acquire adequate training (Kong *et al*, 2000).

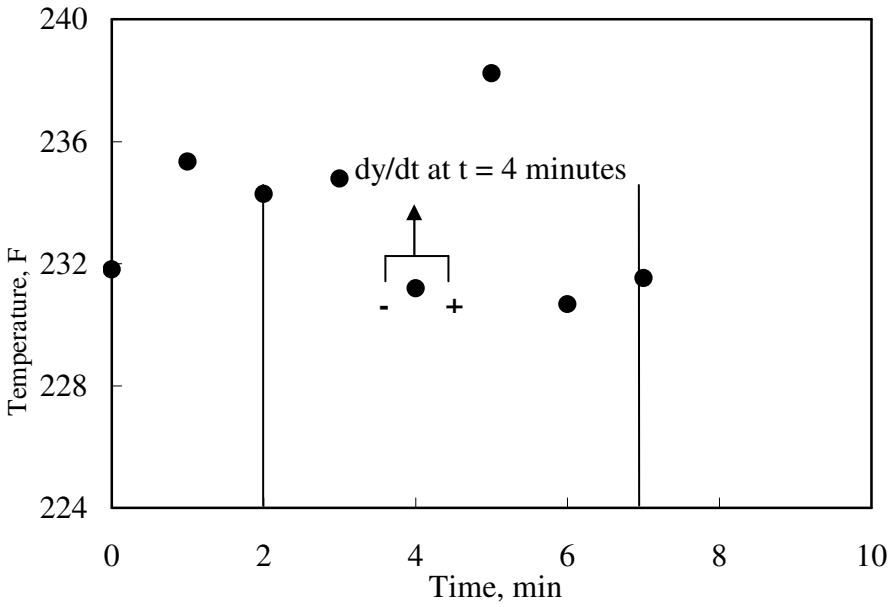
Alici and Edgar (2002) propose a reconciliation method using model identification tools, process simulations and moving window approach. They use model identification tools (e.g.: regression, neural networks etc) to build a simplified version of the model. This simplified version is used instead of the rigorous process model. They report two methods; first using finite differences and the second using time series analysis.

The finite difference based approach converts the equation (II-7) in to equation (II-9). At each sample point in the history horizon, the derivative of the measurement is calculated. This is done by perturbing the process simulation around that time instant (see Figure 2-4). In this fashion, a table is created with the measurement and its corresponding derivative value. From this table of values a relationship in equation (II-9) is developed using either regression or neural networks. The resulting differential equation is converted in to an algebraic equation using the orthogonal collocation method and the reconciliation problem is solved.

$$f\left(\frac{d\hat{Z}(t)}{dt}, \hat{Z}(t), \hat{\theta}(t)\right) = 0 \quad \rightarrow \quad \frac{d\hat{Z}(t)}{dt} = \hat{f}(\hat{Z}(t)) \qquad \text{(II-9)}$$

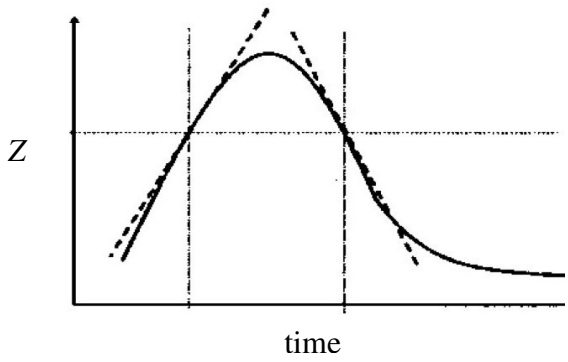
$$f\left(\frac{d\hat{Z}(t)}{dt}, \hat{Z}(t), \hat{\theta}(t)\right) = 0 \quad \rightarrow \quad \hat{Z}(t) = \hat{f}(\hat{Z}(t-1), \hat{Z}(t-2), .., \hat{Z}(t-h)) \qquad \text{(II-10)}$$

Figure 2-4. Calculating Derivate Values Using Finite Difference



In the time series analysis, equation (II-7) is converted to (II-10). Measurements at the current time are expressed as an autoregressive function of all the past *h* measurements, where *h* stands for the history horizon. To estimate this function the process simulation is run several times with starting points along the history horizon. The dynamic response from each run is stored and the resulting table is used to create the time series model. This way the differential equation (II-7) is converted to an algebraic equation and the resulting optimization problem is solved.

Figure 2-5. Limitation of Finite Difference Approach (Figure 2 from Alici and Edgar, 2002)



Finite difference approach used by Alici and Edgar (2002) cannot be used in systems where the relation between the state variable and its differential is of the form shown in Figure 2-5. In this situation one value of measurement can be associated with a positive and a negative derivative. Regression and neural network models are used to express the derivative as a function of the measurement value. These methods cannot work when a given measurement value can correspond to two different derivative value's. This is frequently seen in oscillatory systems (e.g. liquid level in a tank).

Alici and Edgar also report that time series analysis has limited success in tracking fast system dynamics. This is of particular problem when reconciling measurements belonging to process input (e.g. step change in steam flow to a reboiler, feed flow rate etc).

Reconciliation strategies based on model identification tools such as neural networks (Alici and Edgar, 2002; Karjala and Himmelblau, 1996) and regression (Alici and Edgar, 2002) require judicious choice of model parameters. Examples of model parameters are: number of hidden layers in a neural network, model type (i.e. $y = ax + b$ or $y = ax^b$ etc)

in the case of regression. Furthermore, the error from approximating the true process model may corrupt the reconciliation results. Reconciliation strategies based on nonlinear programming (Liebman *et al*., 1991; Albuquerque and Biegler, 1996; Vachhani *et al*., 2005; Haseltine and Rawlings, 2005) based approaches require discretization of the process model and incorporating the resulting model into optimization algorithms. All of the existing methods require significant time and effort to develop rigorous models and the appropriate numerical methods to solve them. The method proposed in the next section minimizes the development effort by utilizing process simulators and optimization software for measurement reconciliation.

3. Rating Based Reconciliation

This chapter presents the reconciliation framework. The engineer using the framework need not program the process equations and the property methods associated with the equations (e.g. thermodynamics and other properties). A dynamic simulation mirroring the process operation is used instead. Building a dynamic simulation takes significantly less effort than programming the model. This framework can also be scaled to situations involving several unit operations.

Process simulations can be built to address two scenarios: Design and Rating. In design mode, the desired performance, e.g. distillate or bottoms composition, is specified and the simulation will calculate the equipment dimensions, e.g. number of trays, required to meet that performance. In rating mode, the equipment dimensions are specified and the resulting performance is evaluated to estimate performance measures, e.g. distillate or bottoms composition. Dynamic simulations are built only in rating mode. Since the method presented here relies heavily on dynamic simulations, it is called Rating Based Reconciliation (RBR).

The RBR framework uses two components:

1. Commercial optimization software that can solve large nonlinear optimization problems (e.g. Premium Solver (Frontline Solvers, 2010) and MATLAB)
2. Dynamic simulation software such as CHEMCAD, Aspen Dynamics and HYSYS.

These two components must be able to interact. In particular, the dynamic simulator must be able to receive input from the optimization software; run simulations based on it and send the results back to the optimization software. This level of interactivity is already present between Excel and several of the simulation programs. For example CHEMCAD allows Excel to access and control simulations using COM interface (Chemstations, 2010). HYSYS simulator can also be similarly controlled by other applications using COM interface (Alici and Edgar, 2002).

COM, Component Object Model, is a Microsoft technology (www.Microsoft.com/COM, March 2011) that allows different software packages to communicate with one another. Using this technology, software developers can allow access to their software features without providing direct access to the code. For example, the VBA code in Figure 3-1 loads a Chemcad simulation that is present on the user's desktop. In lines 3, the specifications associated with the unit operation number 1 in the simulation are accessed and stored in the variable named "spec". Further information on this topic is provided later in this section.

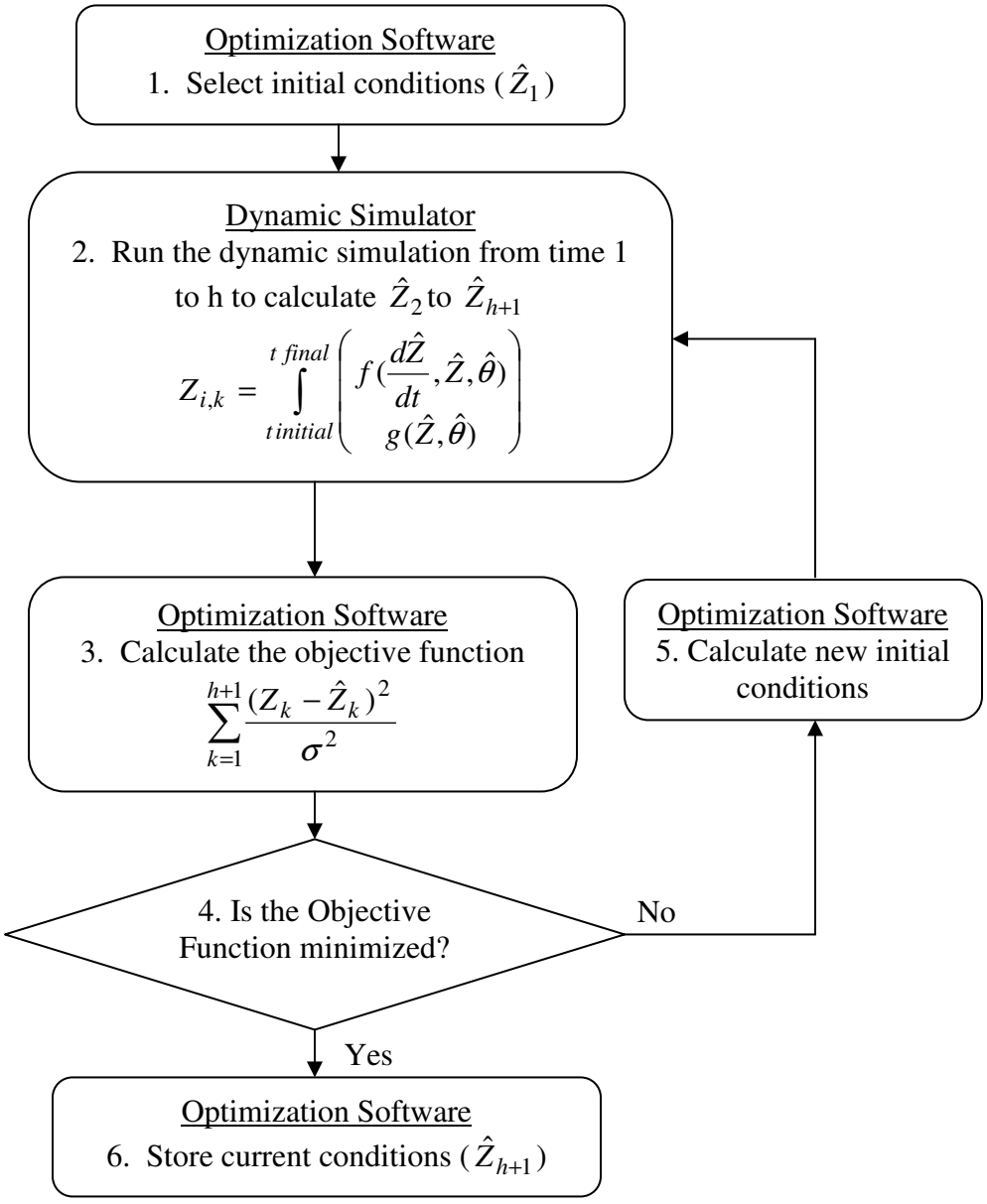
Figure 3-1. Simple Visual Basic Program to Load CHEMCAD Simulation

```
Sub LoadCC5()
      1. Set CC5 = CreateObject("CHEMCAD.VBServer")
      2. CC5.LoadJob("C:\Desktop\CCSimulation.ccx")
      3. Check = unitop.GetUnitOpSpecByID(1, Spec)
End
```

3.1 Algorithm

Reconciliation is a constrained optimization problem (equations (II-6) to (II-8)). RBR method uses the dynamic simulations as a substitute for the process constraints. Since the constraints in the simulator cannot be directly accessed, the method reframes the reconciliation problem as unconstrained optimization problem, equations (III-1) and (III-2). Measurement estimates at the beginning of the calculation horizon ($\hat{Z}_1$) are the optimization variables. These also form the initial conditions for running the dynamic simulation. During reconciliation (Figure 3-2), the optimization software determines the values for the initial conditions. With these initial conditions, dynamic simulation is run to obtain the estimates at the remaining time instances in the calculation horizon. Using these model estimates the objective function value is calculated. This process repeats every iteration until the objective function is minimized. When the objective function is minimized, the model estimates at the current time instance (i.e. last point of the current calculation horizon) are stored.

Figure 3-2. Rating Based Reconciliation Methodology



$$\min_{\hat{Z}_{1,k}} \sum_{i=1}^{n_m}\sum_{k=1}^{h+1}\left(\frac{Z_{i,k}-\hat{Z}_{i,k}}{\sigma_{i,k}}\right)^2 \text{, where } h=\frac{t_{end}-t_{ini}}{\Delta t} \qquad \text{(III-1)}$$

$$\hat{Z}_{i,k} = \int_{t\,initial}^{t\,end} Dynamic\ Simulation \text{ where } 1 < i \le h+1 \qquad \text{(III-2)}$$

h [=] history horizon, n_m [=] number of measurements

In this work, CHEMCAD is used as the dynamic simulation software because of its ease of use and availability at the Department of Chemical & Petroleum Engineering at University of Kansas. MATLAB (Mathworks, 2010) is used for optimization because it allows rapid development of mathematical programs.

MATLAB also provides several optimization routines. Of these, "lsqnonlin" routine is used for reconciliation. This optimization engine is specifically programmed for nonlinear problems where the objective function consists of sum of squares functions. Since objective function in the reconciliation problem is of this form, this routine was used. The optimization is performed using the trust-region-reflective-algorithm (Coleman and Li, 1994 and Coleman and Li, 1996). The algorithm uses sparse linear algebra wherever possible to improve the computation speed (Mathworks, 2010).

The objective function given in equation (III-1) is quadratic. The model may be non-linear in nature. The optimization algorithm will find the globally optimum solution if the model is convex. If the model is non-linear and non-convex then global optimum cannot be guaranteed.

3.2 Interaction Between MATLAB and Excel

CHEMCAD and MATLAB cannot interact with one another directly. However they can interact with Excel. Therefore Excel is used as the interface between MATLAB and CHEMCAD for performing reconciliation. Interactions between MATLAB, Excel and CHEMCAD are programmed using COM technology.

COM, Component Object Model, is an interfacing technology where the components of an application (e.g. first spreadsheet in an Excel file) are exposed to external applications (e.g. a Matlab program) in the form of 'Objects' (i.e. software objects). The components can then be manipulated by the external application. Matlab accesses Excel files using a COM interface called Active X server (Mathworks, 2010). The interaction is initiated and completely controlled by MATLAB. Excel acts as a server and MATLAB is the client. To start the interaction Excel application is run in an automation server process using the `activexserver` function and the program ID `excel.application`.

Figure 3-3. Creating Excel Object in MATLAB

```
excel       = actxserver('Excel.Application');
```

The left side of the above command is the newly created object called `excel`. This object will provide access to all the components of the Excel application. To open an Excel file the `workbooks` method is used. File path is provided as an argument.

Figure 3-4. Commands to access an Excel File from MATLAB

```
wb          = excel.Workbooks.Open('C:\Desktop\Recn.xls');
```

The above command creates an object that links to the selected Excel file. To access a particular worksheet in the Excel file, the `get` method is used with the sheet number as an argument. These commands are shown below.

Figure 3-5. Commands to access Excel Worksheet From MATLAB

```
wb          = excel.Workbooks.Open('C:\Desktop\Recn.xls');
Sheets      = excel.ActiveWorkBook.Sheets;
sheet1      = get(Sheets, 'Item', 1);
invoke(sheet1, 'Activate');
Activesheet = excel.Activesheet;
ws          = excel.ActiveWorkbook.ActiveSheet;
```

The object `ws` provides access to the selected worksheet. To read data from this worksheet, `get` function is used with the location of the cells given as input.

Figure 3-6. Commands to Read from Excel Worksheet from MATLAB

```
location    =   'B2';
Range       =   get (ws, 'Range', location);
DataRead    =   Range.value;
```

The object `Range` provides access to the desired cell whose value is to be read. In similar fashion, to write data to a particular cell in the selected worksheet, get and set functions are used. `get` is used to gain access to a particular cell(s) and `set` is used to specify the value of the accessed cell. In the Figure 3-7, the number 20 is placed in the cell B4.

Figure 3-7. Commands to Write Data to Excel From MATLAB

```
location                = 'B4';
ActivesheetRange        =   get (ws, 'Range', location);
set(ActivesheetRange, 'Value', 20);
```

Using the above discussed commands, MATLAB writes the initial conditions data to an Excel file. The Excel file contains a macro that runs the CHEMCAD simulation using the initial conditions, retrieves the results from CHEMCAD and stores them in one of its worksheets. MATLAB can run the macro using the following command.

Figure 3-8. Command to Run a Macro from MATLAB

```
excel.Run('RunChemcadSimulation');
```

After running the simulation, MATLAB accesses the results from Excel using the above listed commands. The results are then used in calculating the value of the objective function and the Jacobian. These are then used for solving the optimization problem.

3.3 Interaction between Excel and CHEMCAD

An Excel macro is written in Visual Basic that sends the initial conditions to a CHEMCAD dynamic simulation, runs it for a set period of time and writes the results to one its spreadsheets. Here Excel is the master and CHEMCAD is the slave. Initiating the interaction and opening the CHEMCAD simulation are performed using the following commands.

Figure 3-9. Commands to Create CHEMCAD Object and Loading a Simulation from Excel

```
Set CC5 = CreateObject("CHEMCAD.VBServer")
CC5.LoadJob("C:\Desktop\CCSimulation.ccx")
```

The above commands create an object, `CC5`, which provides access to CHEMCAD. In order to write conditions to a stream in the simulation the following steps must be performed (Figure 3-10).

1. Create objects that provide access the flash calculation and stream conditions in the simulation. This is shown in steps 1 and 2
2. The temperature, pressure and component molar flow rates are sent to the CHEMCAD flash calculation to determine the corresponding enthalpy and vapor mole fraction at the specified conditions. To perform the flash calculation the method `calculatetpflash` is used. To access the vapor fraction and enthalpy, the methods `GetMoleVaporFraction` and `GetTotalEnthalpy` are used. These commands are shown in steps 3 to 6.

3. Set the stream conditions with the temperature, pressure, molar flow rates, enthalpy and the vapor mole fraction information. This command is shown in step 7.

Figure 3-10. Setting up Stream Information in CHEMCAD from Excel.

```
1. Set flash = CC5.getflash()
2. Set streams = CC5.getstreaminfo()
3. Check = flash.Definefeedstream(Tempr, Pres, Enth, CompF)
‘ Check will have a value 0 if the operation completed suc-
cessfully
4. Check = flash.calculatetpFlash(Tempr, Pres)
' Below statements get the flashed results from the chemcad.
5. MoleVapFrac = flash.GetMoleVaporFraction()
6. Enth = flash.gettotalenthalpy()
7. StrmSet = streams.PutStreamByID(ID, Tempr, Pres, MoleVap-
```

To access the conditions of any stream the GetStreamByID method is used. This is shown below.

Figure 3-11. Command to access Stream Information from CHEMCAD.

```
Check = streams.GetStreamByID(ID, Tempr, Pres, MoleVapFrac,
Enth, Cflow)
```

To change the specifications of any unit operation in the simulation the following steps are taken (see Figure 3-12).

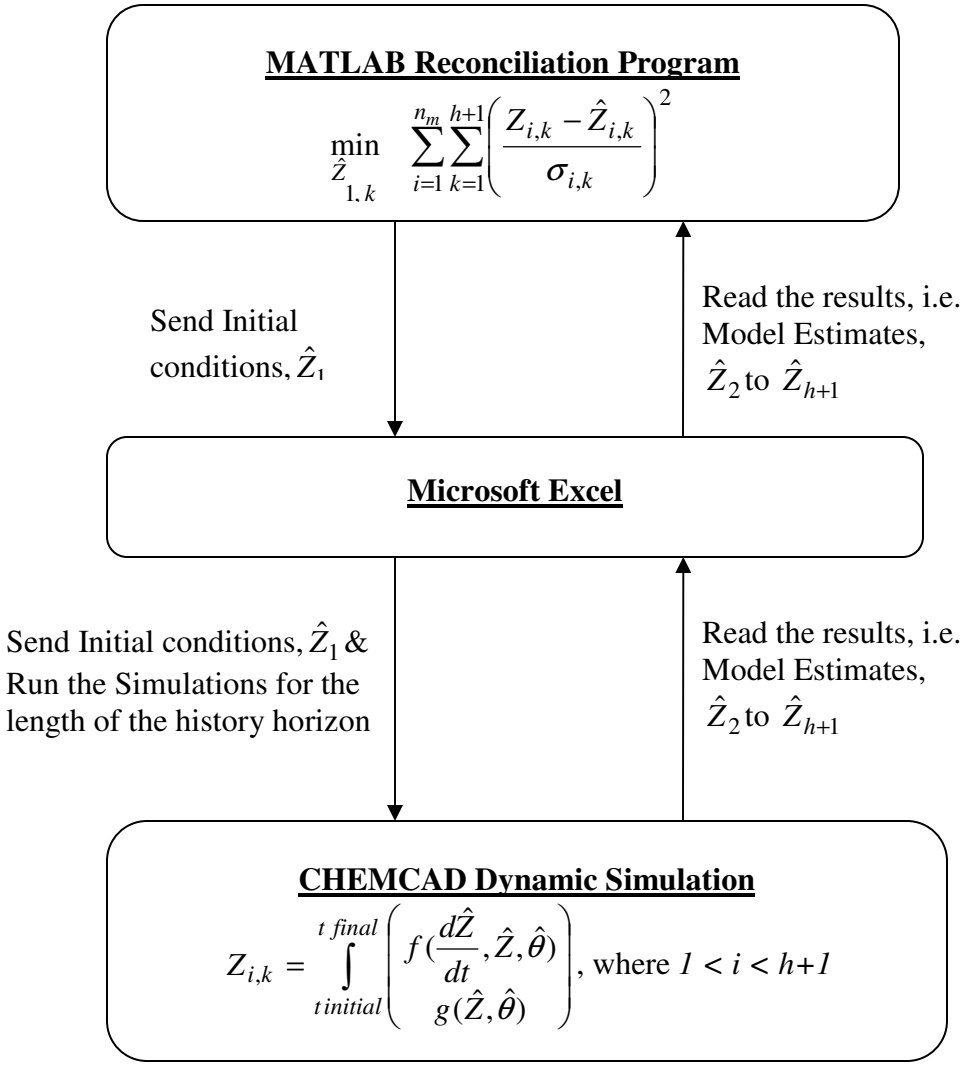
a. Create object that provides access to the unit operations (Step 1).
b. To read the specification of a unit operation, use the `GetUnitOpSpecByID` method. This method requires the unit operation id and returns all the specifications in a vector (Step 2).
c. To change the specification of a unit operation, use the `PutUnitOpSpecByID` method. This method takes the specifications vector as an input and returns zero if successful.

Figure 3-12, Commands to Read and Change Unit Operation Information in CHEMCAD from Excel.

```
1. Set unitop = CC5.GetUnitOpInfo()
2. Check = unitop.GetUnitOpSpecByID(1, Par)
‘ Gets the parameter vector for unit operation # 1.
3. Par(5) = 3
4. Check = unitop.PutUnitOpSpecByID(1, Par)
‘ The above command changes the value of the Par(5) to 3 and
sends this change to the CHEMCAD simulation.
```

The interaction between MATLAB, Excel and CHEMCAD is given in Figure 3-13. MATLAB ←→ Excel interaction is completely controlled by MATLAB. Excel ←→ CHEMCAD interaction is completely controlled by Excel.

Figure 3-13. Interaction Between MATLAB and CHEMCAD via Excel



3.4 Advantages and Disadvantages

3.4.1 Size of Optimization Problem

The size of the optimization problem is much smaller when compared to methods using orthogonal collocation. Orthogonal collocation based methods solve for model estimates through out the horizon ($\hat{Z}_1$ to $\hat{Z}_{h+1}$) and also the collocation between them. RBR method only uses the initial conditions as the optimization variables and therefore the

size of the problem is much smaller. For the dynamic flash tank problem given in Figure 2-3, the number of optimization variables using the proposed RBR method is 28 which is an order of magnitude lower than that from the orthogonal collocation based methods.

3.4.2 Programming Effort

The proposed method requires minimum programming effort from the process engineer when compared to the other methods. The engineer can build dynamic simulations by adding holdup and equipment information to the existing steady state models. This keeps the engineer from having to spend time on programming not only the process equations but also the thermodynamic models required for accurately capturing the process dynamics. Furthermore, this method allows the engineer to use all of the simulation modules which have been developed by others to represent processes behavior.

3.4.3 Computation Speed

The RBR method separates process model and optimization algorithm. Consequently time is spent in transporting significant amount of data between the simulator and the optimization software. This reduces the computational speed of the algorithm.

4. Wavelet Denoising

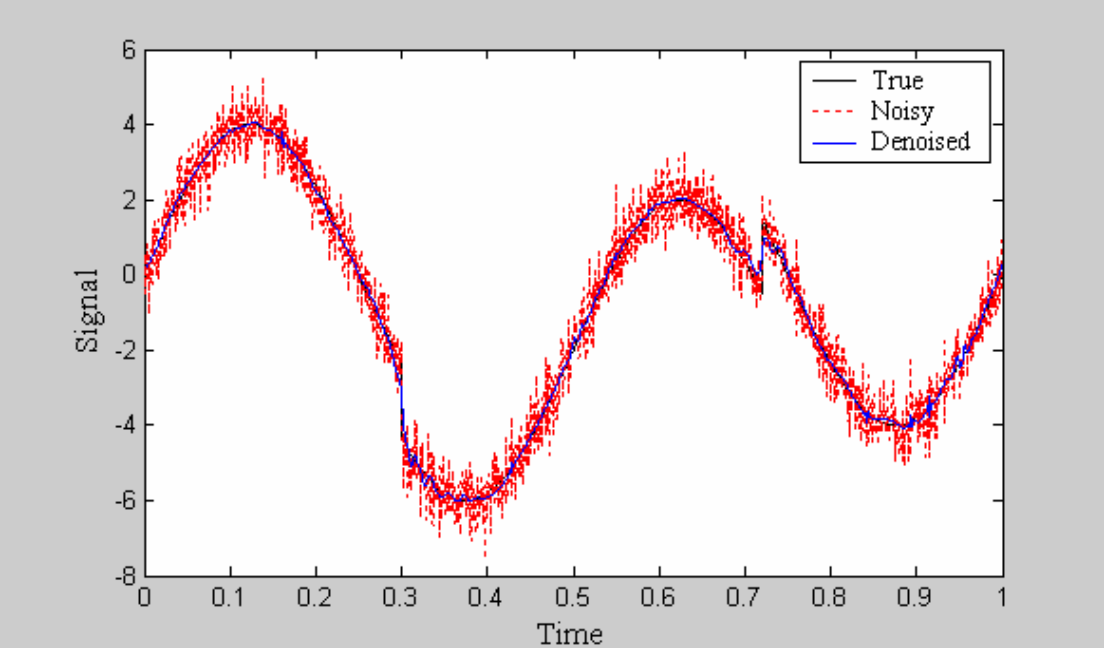
The objective of this work is to develop a method that can properly interpret operating data. The first step is to capture the underlying trends in the dynamic data. The second is to adjust the measurements minimally to close the constraints. The third is to estimate any unknown process variables using the adjusted measurements. All of this is done using optimization software and a process simulator. MATLAB is the optimization software and CHEMCAD is the simulator with Excel facilitating the interaction between these components. This chapter discusses the extraction of underlying trends from noisy data.

MacDonald and Howat (1988) note that reconciliation does not reduce error in all measurements. Lee (2002) and Satuluri (2003) echo the same observation. Process constraints are the only guide for reconciliation to arrive at better measurement estimates. If the constraints are insensitive to a particular measurement, then reconciliation does not reduce the error in that measurement. For example, MacDonald and Howat (1988) report that while reconciling measurements from a flash tank operation, the error in feed pressure measurement is not reduced: the material and energy balances are insensitive to the feed pressure.

Wavelet denoising operates on the raw data without any process constraints invoked. It treats each measurement and its error separately from the rest. Wavelet Denoising attempts to reduce the error by smoothing the measurement trend (see Figure 4-1). In this

fashion it is similar to time averaging filters used in the literature. However there is a key difference. Smoothing filters out any sharp changes in the measurement trend, some of which may be true process changes. Denoising preserves such genuine changes while smoothing the short term fluctuations. Figure 4-1 illustrates this. At time, t = 0.3 and t = 0.7, there are sharp changes in the measurement signal which are genuine. The denoised data retain these changes while filtering the noise during the rest of the time.

Figure 4-1. Illustration of Wavelet Denoising (True Data Generated Using Modified Sine Function from Donoho and Johnstone, 1994)



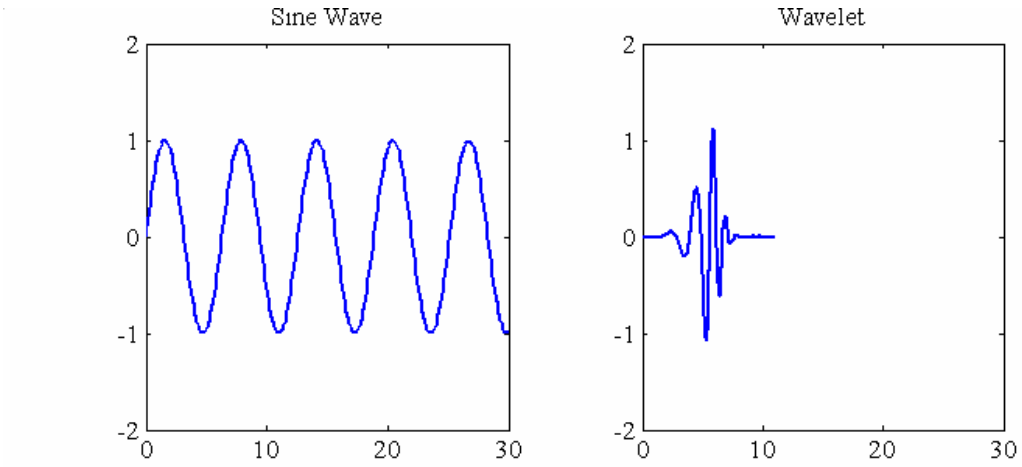
Hinkle (2005) showed that wavelet denoising can be used to complement the RBR method discussed in the previous section. Raw measurements are initially treated using Wavelet Denoising. The resulting denoised measurements are used for reconciliation, i.e. denoised measurements will be used instead of the raw measurements during reconciliation. Since denoised measurements are potentially more accurate than the raw

measurements the overall error reduction in the measurements after reconciliation should increase.

4.1 Wavelet Denoising Algorithm

The field of wavelets was initiated by Morlet and Grossman in 1980 and later developed by Mallat, Meyer and Daubechies (Hubbard, 1998). Much of the wavelet analysis is through wavelet transforms. The idea behind wavelet transforms is similar to that of Fourier transforms. In Fourier transform a function is represented as a combination of sine and cosine functions of various frequencies. Similarly in the wavelet transform a combination of wavelets with various frequencies are used to represent the measurement signal. The key difference is that the sine's and cosine's functions are of infinite duration whereas wavelets are of finite duration (Figure 4-2). This property makes wavelets useful for analyzing real world signals/data of finite duration and sharp discontinuities (Graps, 1995; Hubbard, 1998).

Figure 4-2. Sine Wave and a Wavelet



The wavelet denoising method can reduce error when the noise frequency is higher than the changes in the process information frequency. The goal of the method is to remove the high frequency variations and leave the low frequency content. The denoising procedure consists of the following steps (Taswell, 2000):

1. Perform discrete wavelet transform of the data to divide it into high frequency and low frequency content.
2. Filter the high frequency content via Thresholding.
3. Rebuild the data by performing inverse wavelet transform on the filtered signal.

4.1.1 Discrete Wavelet Transform

The discrete wavelet transform decomposes the signal into two subsignals, each roughly half its size (Figure 4-3 and Figure 4-4). The first, called the approximation, represent the gross features (i.e. low frequency content) in the signal. The approximations are akin to running averages. The second called detail, show the finer changes (i.e. high frequency content) in the signal. The details are similar to running differences. If the original signal contained n data points, then the Approximation and the Detail signals will roughly contain *n/2* points each.

The noise present in the original signal is partly captured in the Detail sub signal. The noise remaining in the Approximation sub signal can be further separated by applying discrete wavelet transform on it. The subsignals resulting from this second wavelet decomposition are called approximation 2 (A2) and detail 2 (D2). The number represents

the wavelet decomposition level. By repeating this process a few times, most of the noise in the data will be captured in the detail components (D1, D2, D3 etc).

There is a limit to the number of times a signal can be decomposed. If the signal is exactly halved each time it is decomposed, then the maximum decomposition level is $\log_2 N$. However the signal is not exactly halved during the decomposition. The maximum decomposition level is given by equation (4-1) (Misiti *et al*, 1993). It is dependent on the wavelet used for the decomposition. The maximum decomposition level reduces as the number of points in the wavelet increases.

$$\text{Maximum Decomposition level} = \frac{\log \frac{N}{N_w - 1}}{\log 2} \qquad (4\text{-}1)$$

Where N is the number of data points in the original signal.

N_w is the number of points in the wavelet.

As the signal is decomposed to its maximum level, the approximation sub signal may lose process information along with the noise (Hinkle, 2005). Hence, the signal is not decomposed to its maximum level. MATLAB Wavelet toolbox is used for denoising in this work. In MATLAB, the signal is usually decomposed five times (Mathworks, 2010).

4.1.2 Thresholding

The noise captured in the details is removed in Thresholding. Usually this noise has the lowest magnitude. To remove the noise, all the data in the details (D1 to D3) which has a magnitude less than a threshold value (λ) are deleted. This is called Hard-Thresholding (Figure 4-5). An alternative is to shrink the data in the details by a value equal to the threshold. This is called Soft-Thresholding (Figure 4-6). Jansen (2001) states that Soft-Thresholding is preferred because it preserves continuity in data while damping the fluctuations.

Figure 4-3. Division of the Signal into Approximation and Detail (MathWorks, 2010)

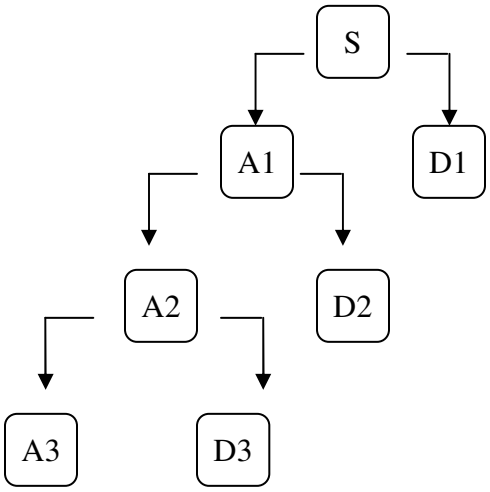


Figure 4-4. Discrete Wavelet Transform

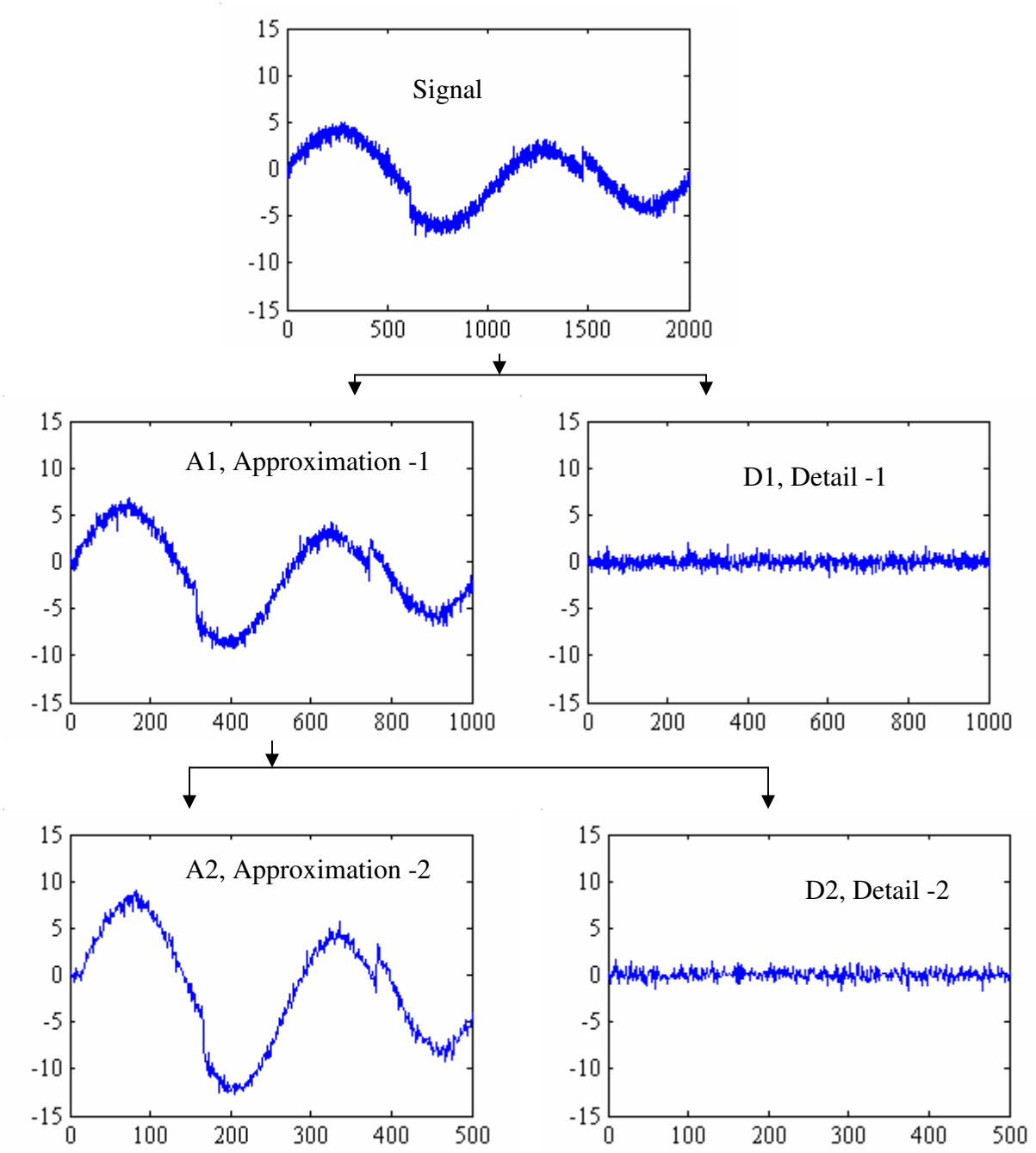


Figure 4-5. Hard-Thresholding

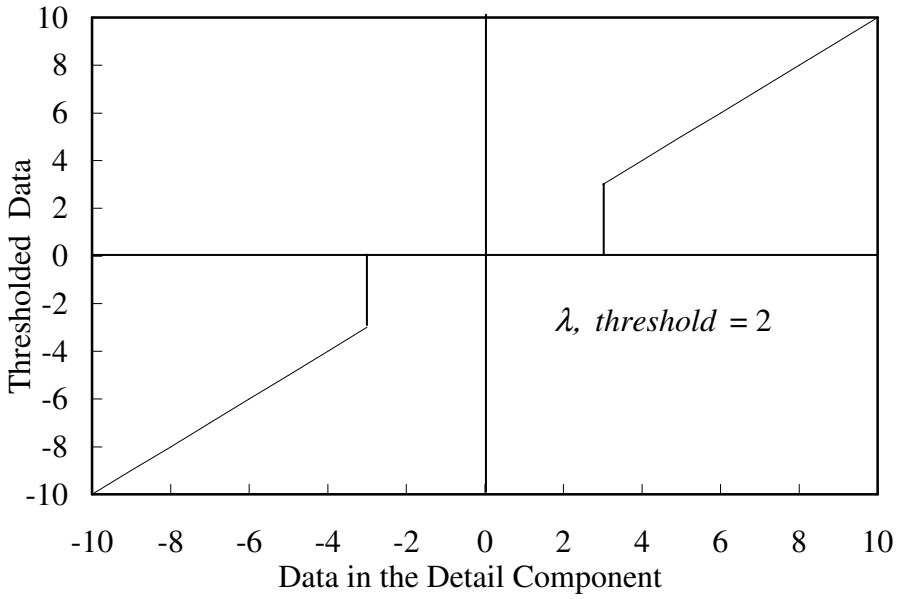
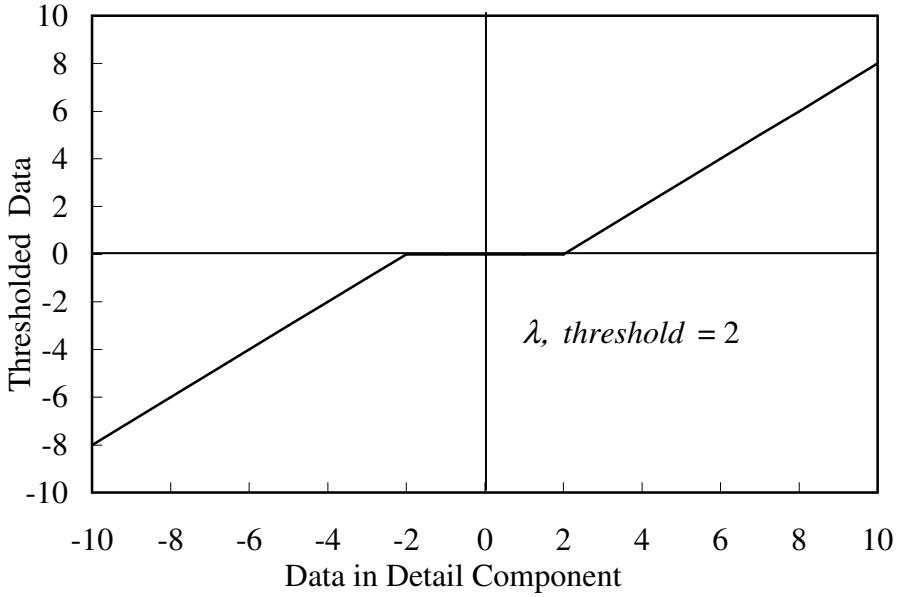


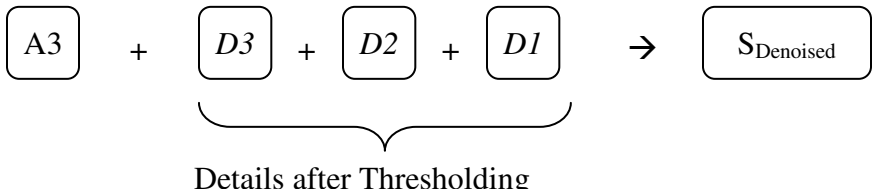
Figure 4-6. Soft-Thresholding



4.1.3 Inverse Wavelet Transform

After applying Threshold to the detail components of the signal, inverse wavelet transform is performed. In the inverse wavelet transform, the truncated data in the details (*D1* to *D3*) along with the approximation (A1) is used to reconstruct the signal (Figure 4-7). Since the noise present in the details is filtered, the rebuilt signal will also have reduced fluctuations.

Figure 4-7. Inverse Wavelet Transform



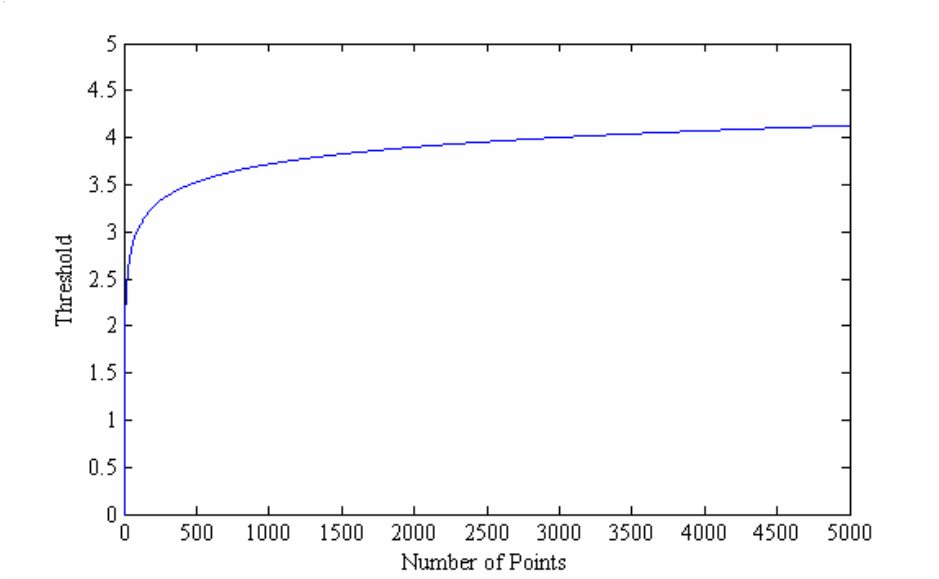
4.2 Threshold Selection

The selection of the threshold is important for wavelet denoising. It has to be sufficiently large to ensure that the noise is removed while it also has to sufficiently small to make sure that any fluctuations that are not noise are not deleted. There are two classes of methods available for calculating the thresholds.

4.2.1 Universal Threshold

Donoho and Johnstone (1994), proposed a simple method for calculating the threshold based on the number of sample points present in the data. The threshold is equal to $\lambda_{UNIV} = \sqrt{2 \cdot \log N} \cdot \sigma$ (Jansen, 2001). While it seems unreasonable to base the threshold levels on the number of data points, Jansen (2001) argues that in the wavelet domain it makes sense. He states that the threshold is very weakly dependent on the number of points (see Figure 4-8).

Figure 4-8. Effect of number of points on Universal Threshold.

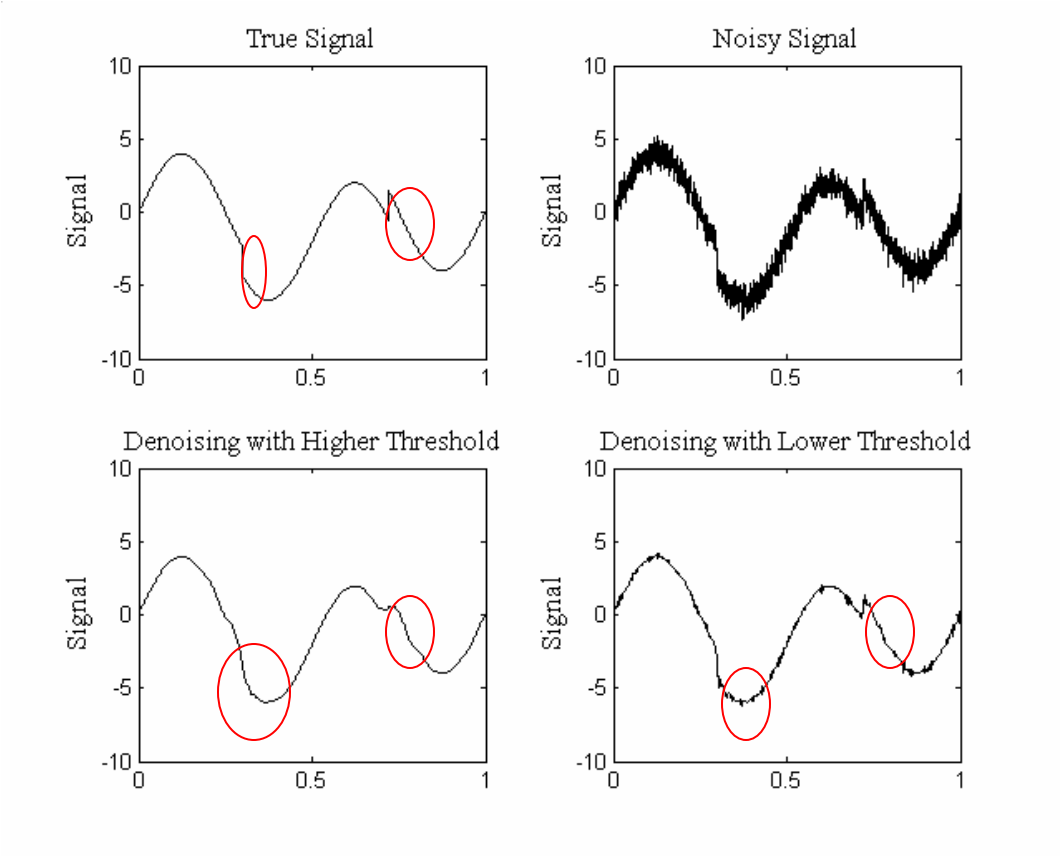


4.2.2 Data Adaptive Threshold

Donoho and Johnstone (1995) proposed a threshold calculation method that is dependent on the data and not on the number of points. The idea behind this method is as follows. Let bias be defined as the difference between the denoised signal and the original signal and variance defined as the measure of variation in the denoised signal. The threshold value selected affects the variance and bias in opposite ways. Increasing the threshold increases the amount of variation that is removed (as more detail coefficients are set to zero). However, increasing the threshold also increases the likelihood that any real variation in the signal captured by the detail coefficients may be lost in the denoising. As a result with increasing threshold value, losing the true variation in the signal due to denoising, increase the bias from the procedure. In other words, increasing threshold reduces the variation (i.e. more smoothing) at the expense of introducing bias.

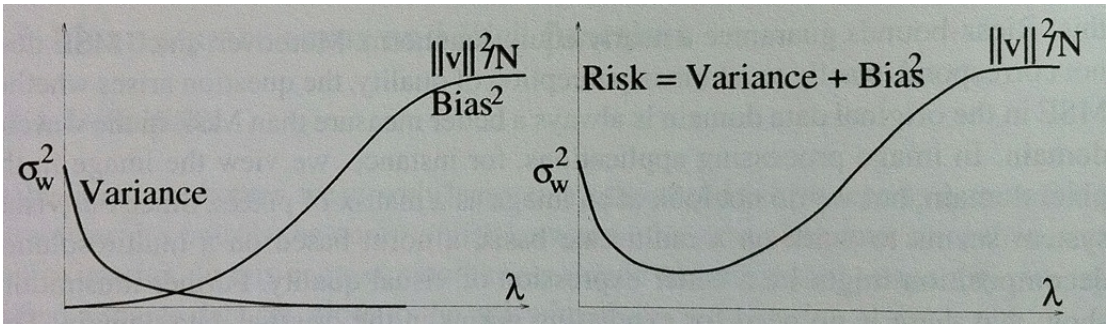
Figure 4-9, shows the impact of threshold value on denoising. Comparing the two plots in the second row of the figure, one can see that using higher threshold leads to smoother version of the signal. However, this smoothness is obtained at the expense of neglecting the true discontinuities at highlighted portions of the plot. The bottom right plot, using lower threshold value, does not result in a smooth curve; however it preserves the true discontinuities in the original signal.

Figure 4-9. Effect of Threshold value on Denoising.



If risk were defined as a combination of variance and bias, then the effect of threshold value on the risk is shown in Figure 4-10. There is an optimum threshold value that minimizes the variance as well as the bias. However, in reality bias and variance cannot be calculated as the true signal is unknown. Donoho and Johnstone (1995) proposed that while this optimum value cannot be calculated, it can estimated just as population mean is estimated by the average. They propose using Steins Unbiased Risk Estimate for estimating the risk. They call their method, SURE, after the estimation type they use.

Figure 4-10. Effect of Threshold Value on the Risk (Picture from Jansen, 2001)



Jansen (2001) states that 'SURE' is better than Universal Threshold for threshold calculation. However, if the signal to noise ratio is small, then threshold calculated using SURE will result in very noisy denoised signal (Mathworks, 2010). Under such circumstances, it is better to use Universal threshold. Matlab is used for denoising in this work. The software provides an option that uses Universal threshold under low signal to noise ratio and SURE method otherwise. This option is used in this work.

4.3 Wavelet Selection

Wavelets useful for denoising must meet the following criteria (Addison, 2002; Fugal, 2009).

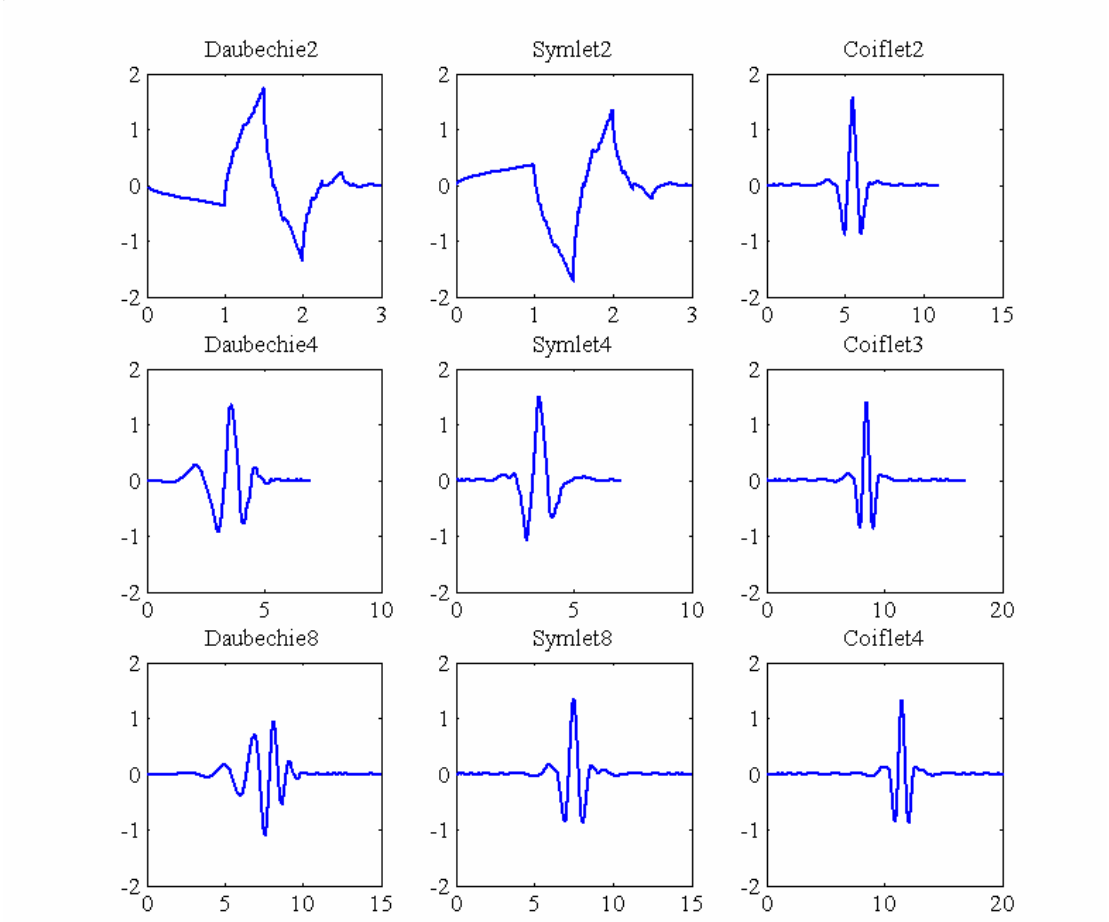
1. Zero mean.
2. Finite duration.
3. Perfect Reconstruction.
4. Orthogonal representation.

There are several wavelets available in the literature (e.g. Mexican Hat, Morlet, Gaussian, Shannon etc). Only few of them meet the criteria listed above. Matlab offers three such wavelet families which are powerful and widely used (Fugal, 2009). These are: Daubechies, Symlets & Coiflets. Each family consists of several wavelets. They are numbered sequentially, e.g., Daubechies 1 to 45, Symlets 1 to 45 and Coiflets 1 to 5. The number assigned to each wavelet is proportional to the number of vanishing moments associated with the wavelet (Mathworks, 2010). If a wavelet has vanishing moments of *n* it means that it can reproduce polynomial signals of order *n-1* (Mohlenkamp and Pereyra, 2008).

Figure 4-11 shows examples of Daubechies, Symlets and Coiflets. They were all designed by Ingrid Daubechies (Fugal, 2009). As seen in the figure, they are of finite duration and have zero mean. They have proven orthogonality and perfect reconstruction properties (Fugal, 2009). As the number of vanishing moments of a wavelet increases, the complexity of its shape also increases. The amplitude shortens and the oscillations tend to get narrower. These are common to all of the wavelet families.
Daubechies wavelets were the first wavelets that were used for denoising (Donoho, 1993). They had the shortest duration when compared to other wavelets with equal number of vanishing moments. The computation time associated with the wavelet transforms is proportional to their duration. Since Daubechies wavelets had the shortest duration, they also offered quick computation times. Symlets were designed to be more symmetrical than the original Daubechies wavelets. They find use in image processing as the human eye is more forgiving of symmetrical errors (Fugal, 2009). This work uses Daubechies wavelets because they are robust, fast and adaptable (Fugal, 2009; Hinkle,

2005). They perform well for analyzing signals with slow dynamics (e.g., liquid level in a large storage tank), fast dynamics (e.g., pressure in a steam header network) and dynamics in between.

Figure 4-11. Comparison of Wavelets



The Daubechies family consists of 45 wavelets. They differ from one another by the number of vanishing moments (herein referred to as wavelet order). To select the wavelet used in this work, a simulation test was performed. Two measurement trends typically

seen in chemical processes, but with different characteristics were denoised. The first measurement trend is that of a scaled concentration in a CSTR and the second is scaled feed concentration to the same CSTR (see Figure 4-12). The reactor concentration changes slowly whereas the feed concentration undergoes a step increase at t = 31 seconds. These two measurement trends represent two extremes seen in the process data: slow changing (reactor concentration) and quick changing (feed concentration). Wavelet order that performs well for these two extremes will be used in this work. Daubechies wavelets from order 2 to 10 are used to denoise the measurements (There is no Daube-chie wavelet of order 1).

For comparing the performance of the various wavelet orders the following statistic was used.

RMSE [=] Root Mean Square Error

[=] $\sqrt{Variance + (Bias)^2}$

Where,

Error [=] | Measured Value – True Value |

Variance [=] Variance in the error

Bias [=] Average Error

The error in the measurements and the denoised measurements is quantified by RMSE. The percentage drop in RMSE after denoising is shown in Figure 4-13. It compares the RMSE reduction in the reactor concentration and feed concentration measurements when

using Daubechies 2 to Daubechies 10 wavelets. Figure 4-15 and Figure 4-16 compare the trends in the true, measured and denoised data for the various wavelets.

Lower order Daubechie wavelets reduce RMSE significantly when the underlying signal is changing slowly with time. The reactor concentration changes slowly with time. The highest RMSE reduction for the reactor concentration is via Daubechies 2 wavelet. It incrementally reduces as the wavelet order increases to Daubechies 10 (Figure 4-13). The reason can be seen in Figure 4-16. Using the lower order Daubechies, much of the fluctuations are filtered in the denoised data. As the wavelet order increases, the de-noised data retain more and more of the fluctuations in the measurements.

Higher order Daubechie wavelets perform better than the lower order, when the underlying signal contains sharp changes. Lower order wavelets try to smooth the sharp changes (Figure 4-15). The error in the feed concentration measurements increases after denoising when using the lower order wavelets for this reason. Consequently the reduction in RMSE is negative (Figure 4-13). Higher order wavelets preserve the step change in the feed concentration.

The wavelet used for measurement denoising must be able to recognize and respect sharp changes while filtering smaller noise fluctuations. Daubechie 4 wavelet strikes a good balance in this regard (Figure 4-15 and Figure 4-16). Figure 4-14 shows the reduction in RMSE, when the area around the step change is excluded from the analysis. Here it can

be seen that Daubechie 4 wavelet performs the best in denoising the reactor and the feed concentrations. It is used in this work.

The choice of the best wavelet may change depending on the speed and size of the changes seen in the process and the magnitude of the noise when compared to the measurement value (or the signal). The speed and size of the changes are affected by the changes in the residence time of the process. For example, reduction in the liquid level or increased throughput will affect the dynamics of the process and the performance of the wavelet.

There is a need for a methodology that can adaptively select the appropriate wavelet taking into account the current process dynamics. When there are fast process changes, higher order wavelets can be selected and during slow process behavior lower order wavelets are chosen. While shifting from one wavelet to another, continuity must also be preserved.

Figure 4-12. Reactor and Feed Concentration to a CSTR

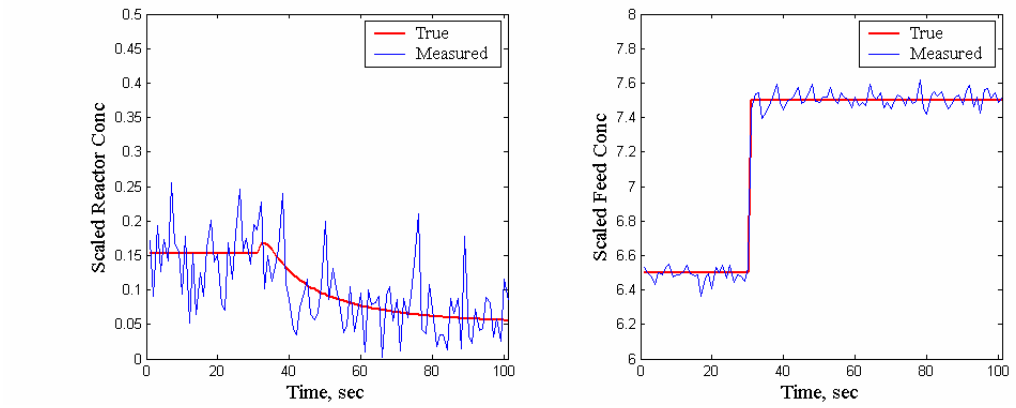


Figure 4-13. Percentage Reduction in RMSE Via Denoising

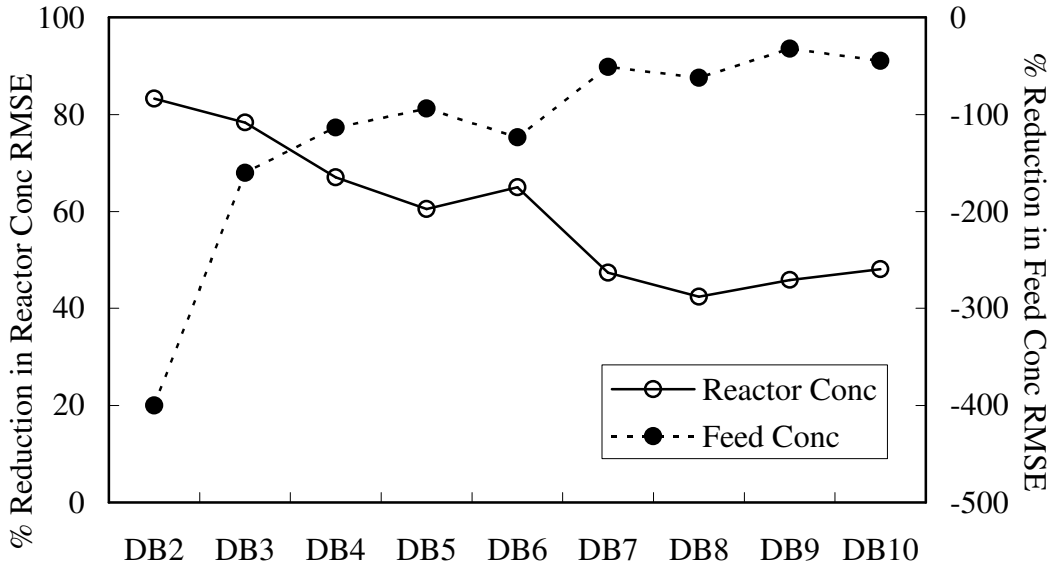


Figure 4-14. Comparison of RMSE (Excluding t = 20 to 50 seconds, i.e. Step Change Area)

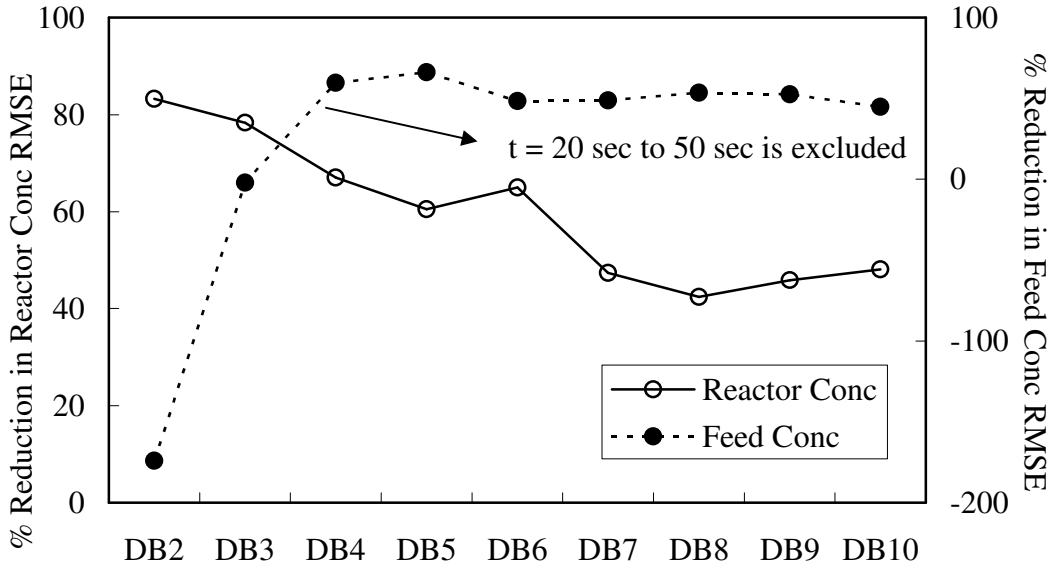


Figure 4-15. Denoising of Scaled Feed Concentration

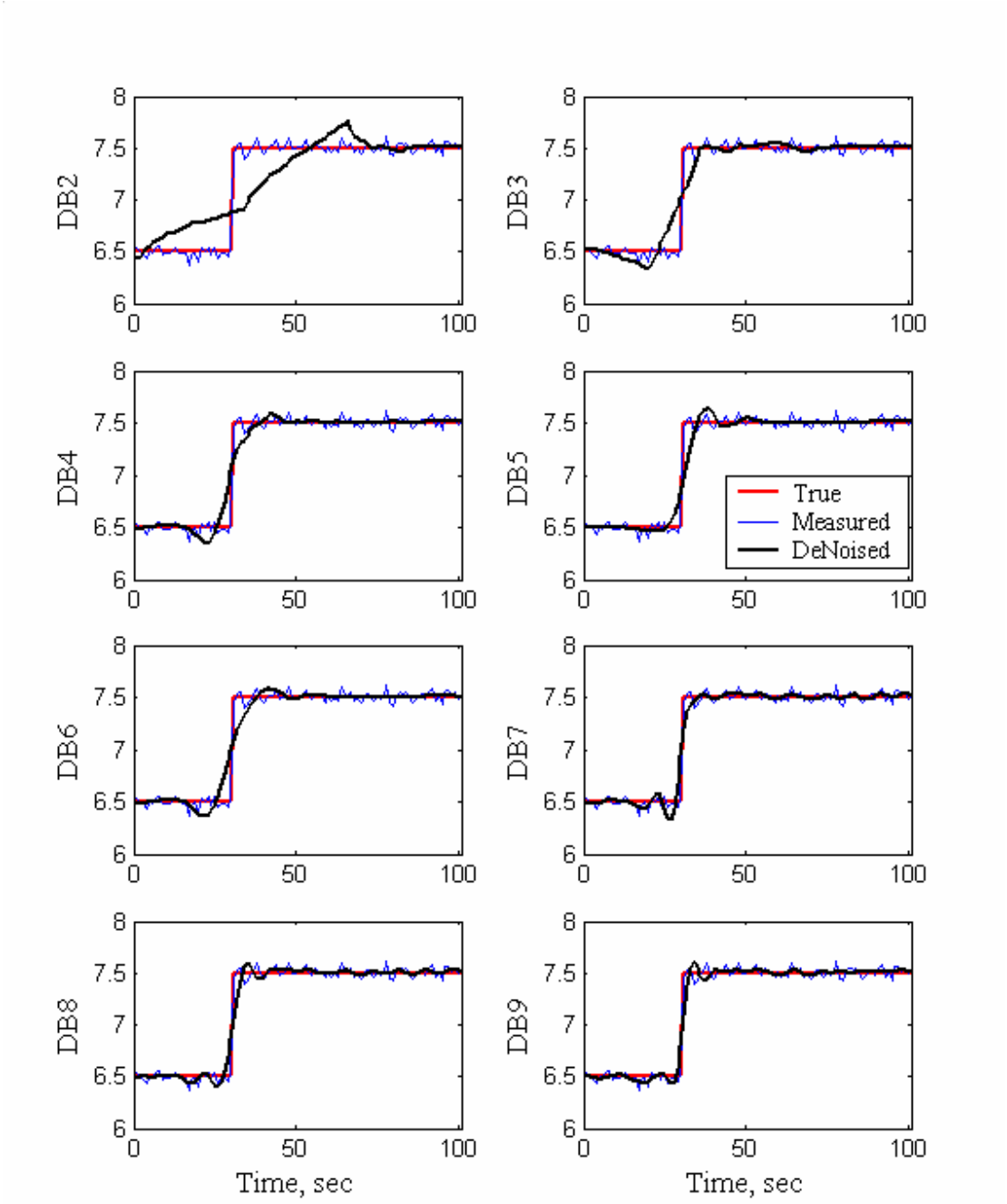
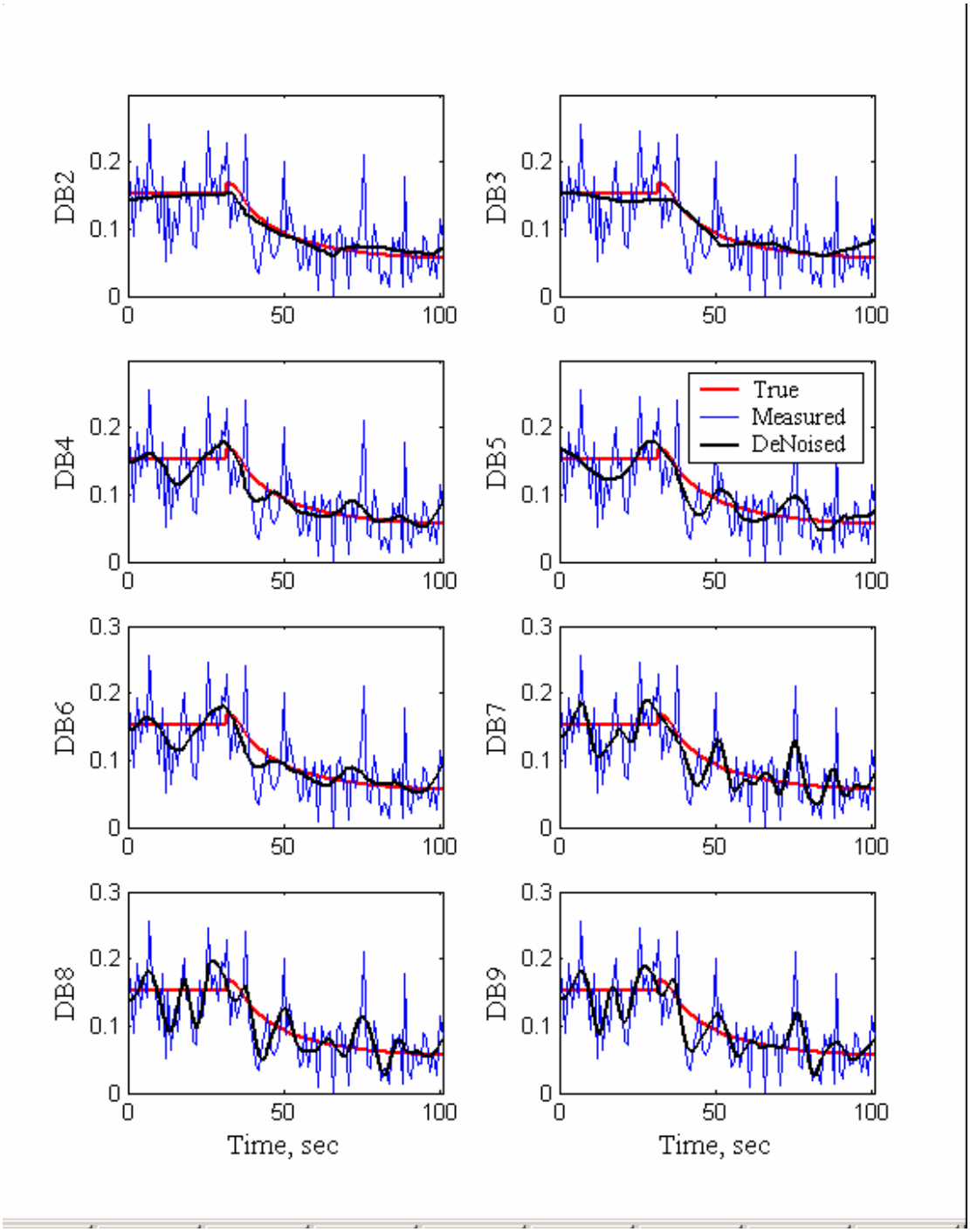


Figure 4-16. Denoising of Scaled Reactor Concentration.



5. Results and Discussion

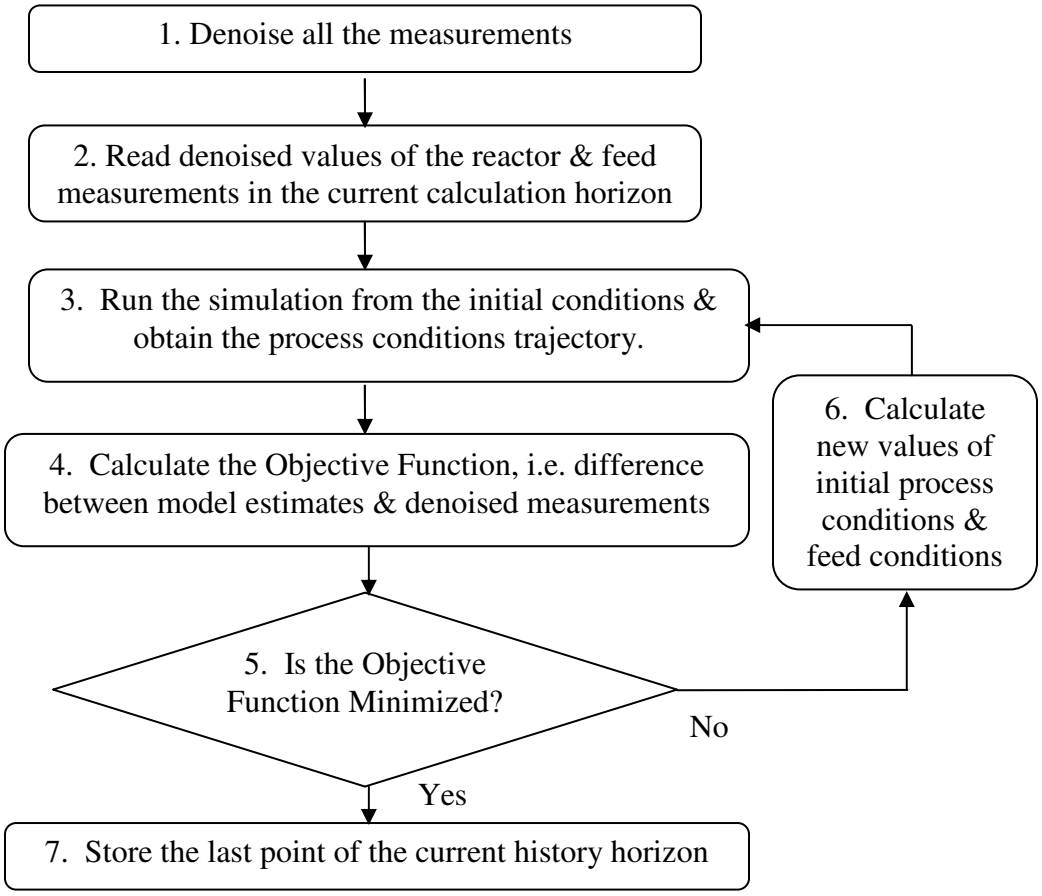
Three case studies in increasing complexity are presented where Wavelet Denoising and Rating Based Reconciliation are used to improve the process data. The results show that the methods successfully improve the measurement accuracy and estimate unmeasured variables while satisfying the process equations.

5.1 Algorithm and Performance Metrics

5.1.1 Algorithm

Measurements are denoised and reconciled, following the procedure in Figure 5-1, for each time step in the moving horizon. The starting conditions in the current horizon are the optimization variables. They are adjusted to minimize the difference between the model predictions and the corresponding denoised measurements from the current horizon. Once the objective function is minimized, the model predictions become the reconciled measurements. The latest of these reconciled measurements along with any estimated variables are stored. The process is then repeated for the next horizon.

Figure 5-1. Rating Based Reconciliation Algorithm

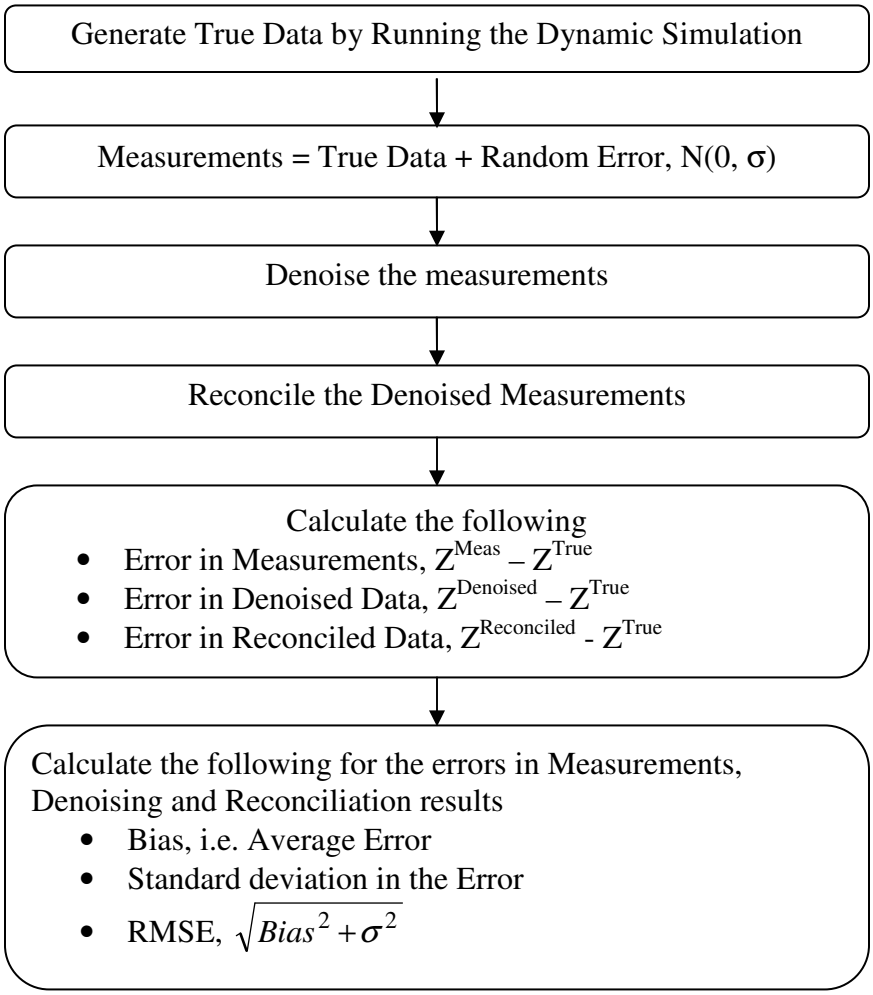


5.1.2 Testing and Performance Metrics

The testing algorithm is shown in Figure 5-2. Process measurements are simulated values which represent the true behavior. Measured values are developed by adding random error to each measurement assuming normal distribution with zero mean (no bias) and standard deviation set to the measurement uncertainty.

The statistical deviations between the measured, denoised and reconciled values from the true one represent the error and provide insight into the strength of the denoising and RBR procedures. The statistical measures are standard deviation, bias and root mean square error.

Figure 5-2. Reconciliation Testing Procedure



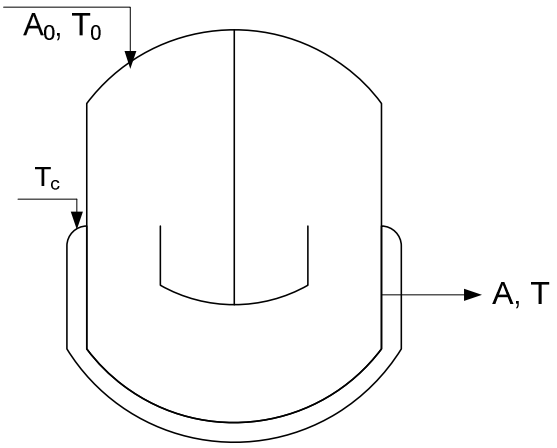
5.2 Simple Jacketed Reactor

5.2.1 Background

This case study consists of a jacketed CSTR with a first order exothermic reaction containing one reactant and product. This example was first formulated by Seinfeld (1970) and was later used as a reference example by several authors (Jang *et al*, 1986; Liebman *et al*, 1992; Alici and Edgar, 2002). It serves as a good case, the results of which can be used for comparison.

A schematic of the reactor is given in Figure 5-3. The heat from the reaction is continuously removed by the coolant in the jacket. The reaction is A → B. The model consists of equations (V-1) and (V-2).

Figure 5-3. Jacketed Reactor



$$\frac{dA}{dt} = \frac{q}{V}(A_0 - A) - (\alpha\, k\, A) \qquad \text{(V-1)}$$

$$\frac{dT}{dt} = \frac{q}{V}(T_0 - T) + \alpha\left(\frac{-\Delta H}{\rho C_p}\right) k\, A - \frac{U(Area)}{\rho C_p V}(T - T_c) \qquad \text{(V-2)}$$

Where, A_0 [=] Concentrations of component A in the feed (gmol/cm^3).

A [=] Concentration of component A in the reactor (gmol/cm^3).

α [=] Catalyst deactivation parameter. It varies from 0 to 1.

C_p [=] Specific heat capacity of component A (cal/g.K).

ΔH [=] Heat of reaction (cal/gmol).

k [=] Temperature dependent reaction rate coefficient.

q [=] Flow rate of the feed and exit streams (cm^3/min).

ρ [=] Density of the feed and exit streams (g/cm^3)

V [=] Reactor holdup volume (cm^3).

T, T_0, T_c [=] Feed and coolant temperatures (K).

U [=] Heat transfer coefficient of the jacket (cal/(cm^2.sec.K)).

The assumptions behind these equations are:

- Holdup volume of the reactor is constant.
- Incoming and outgoing mass flow rates are equal.
- Coolant flow rate is high enough such that there is insignificant change in coolant temperature.
- Density and specific heat capacities of the streams do not change with composition.
- The work of agitation is negligible.

The model equations given by (V-1) & (V-2) are normalized by dividing the concentrations and temperatures (e.g. A` and T`) with reference concentration A_r and reference temperature T_r. The notation used here is same as that used by Liebman *et al* (1992). The equations after normalization are given below:

$$\frac{dA'}{dt} = \frac{q}{V}\left(A'_0 - A'\right) - \left(\alpha\, k\, A'\right) \qquad \text{(V-3)}$$

$$\frac{dT'}{dt} = \frac{q}{V}\left(T'_0 - T'\right) + \alpha\left(\frac{-\Delta H A_r}{\rho C_p T_r}\right) k\, A - \frac{U(Area)}{\rho C_p V}\left(T' - T'_c\right) \qquad \text{(V-4)}$$

where

$k = k_0\, e^{\frac{-E}{T' T_r}}$, E [=] activation energy / gas constant (K) (V-5)

The model consists of four variables: feed concentration (A_0), feed temperature (T_0), reactor concentration (A) and reactor temperature (T). The reactor would operate in unsteady state when the feed conditions change. In the following pages measurements from such transient conditions are reconciled using the RBR method.

Table 5-1. Constants Used in the Model

Symbol	Value
A_r	10^{-6} gmol/cm^3
Area	10 cm^2
C_p	1 cal/g K
E	14090 K
k_0	7.86 x 10^{12} sec^{-1}
ΔH	-27,000 cal/gmol
q	10 cm^3/s
ρ	0.001 g/cm^3
T_c	340 K
T_r	100 K
U	0.0005 cal/(cm^2 sec K)
V	1000 cm^3

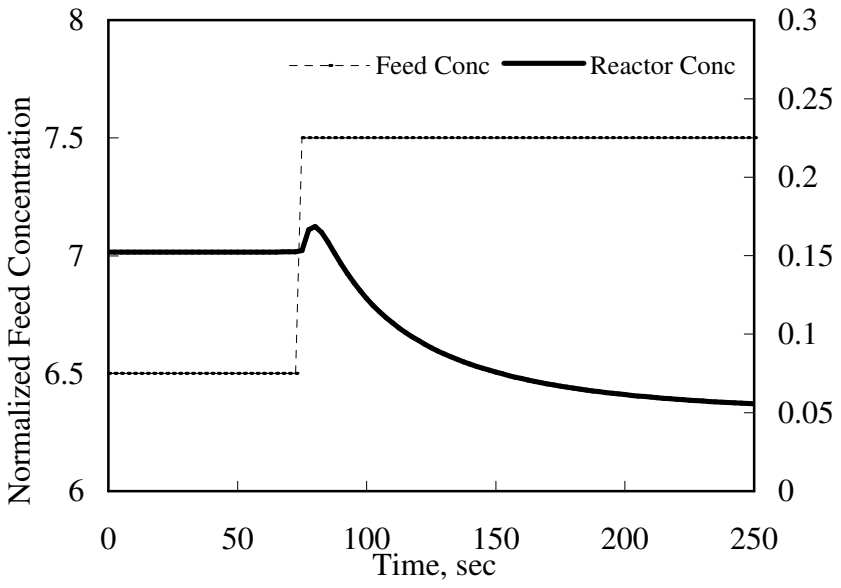
A process model was built using MATLAB. This model solves the model equations (V-3) and (V-4) using semi-implicit Runge-Kutta method (Holland and Liapis, 1983). Since the model equations are a simple set of differential equations it is easier to solve them numerically than to create a model in Chemcad with chemical components that meet the properties listed in Table 5-1 and also satisfy the assumptions. Therefore, the process model is the process simulation for this problem.

5.2.2 Results without Denoising

Liebman *et al*. (1992) and Alici and Edgar (2002) use a step change in feed to test their methods. These conditions are shown in Figure 5-4. Liebman *et al*. (1992) reconcile by converting the differential equations into algebraic ones via the Method of Weighted Residuals (MWR). Alici and Edgar (2002) use process simulations to build simplified models of the process and then use MWR. RBR is used to analyze the same case study and to allow direct comparison to results from their methods.

Measurements were sampled every 2.5 seconds. Size of the history horizon is 12.5 seconds. Data from the past five sampling instances are used in reconciling the current measurement. In order to evaluate the performance of just the reconciliation, measurements were not denoised. The next subsection shows the results using denoising.

Figure 5-4. Feed and Reactor Concentration Trends



The reconciliation results are given in Table 5-2. There is significant reduction in the RMSE of the reactor conditions and feed temperature measurements. Figure 5-7, Figure 5-9 and Figure 5-11 show the trends in the measured, reconciled and true data. The reconciled values have much less scatter and are closer to the true values when compared with the measurements. This is reflected as the reduction in RMSE.

The standard deviation and RMSE in the feed concentration increase after reconciliation (Table 5-2). The reconciliation algorithm assumes that the feed conditions stay constant during the calculation horizon and that it is equal to the initial feed conditions. This is a valid assumption for the most part. However, when the current horizon involves the step change in the feed concentration, it becomes invalid. The reconciliation results show a delay in recognizing the step increase in the feed concentration (see Figure 5-5). Consequently it introduces bias in the results, which is reflected in the results. To verify, the error in feed concentration was calculated by excluding the region that involves the step change. Under such circumstances the RMSE in the feed concentration reduces due to reconciliation (see A_0^* in Table 5-2).

Liebman *et al*. (1992) and Alici and Edgar (2002) assume constant feed conditions during reconciliation calculation. They do this to minimize the number of optimization variables. They, too, report increase in the error for feed concentration from this assumption. Constant feed concentration was assumed in this work to ensure a fair comparison to the results from Liebman *et al*. (1992) and Alici and Edgar (2002). It is not due to any limitation in the RBR method.

Table 5-2. Reconciliation Results for a Step Increase in Feed Concentration.

	Measurements		Reconciliation		% Drop in
Variable	Bias	Std Dev	Bias	Std Dev	RMSE
A	-0.002	0.047	-0.001	0.011	77.3
T	0.000	0.054	0.001	0.016	70.5
T_0	-0.005	0.046	-0.006	0.021	53.2
A_0	0.002	0.045	-0.024	0.126	-185.4
A_0^*	0.003	0.043	0.002	0.016	61.4

A_0^* -Feed concentration excluding the region of step change (from t = 72.5 to 95 seconds)

The standard deviation reduction from this work, Liebman *et al*., (1992) and Alici and Edgar (2002) are listed in Table 5-3. Results from this work are better than those from Alici & Edgar. Results from Liebman *et al*. are better than that from this work. Figure 5-5 to Figure 5-12 show the actual data from this work and Liebman *et al*. Liebman *et al*.'s results are reported in time steps. One time step is equal to 2.5 seconds and is equal to the time interval used in this work. Qualitatively, the reduction of scatter and the proximity of reconciliation results to the true data look equivalent for both methods.

Table 5-3. Comparison of Reconciliation Results

	Percentage Reduction in Std. Dev		
	This Work	Liebman *et al*. (1992)	Alici & Edgar (2002)
A	77.4	87.8	62.0
T	70.6	77.1	54.0
A0	-	-	-
T0	54.5	65.7	44.0

Figure 5-5. Reconciliation of Feed Concentration

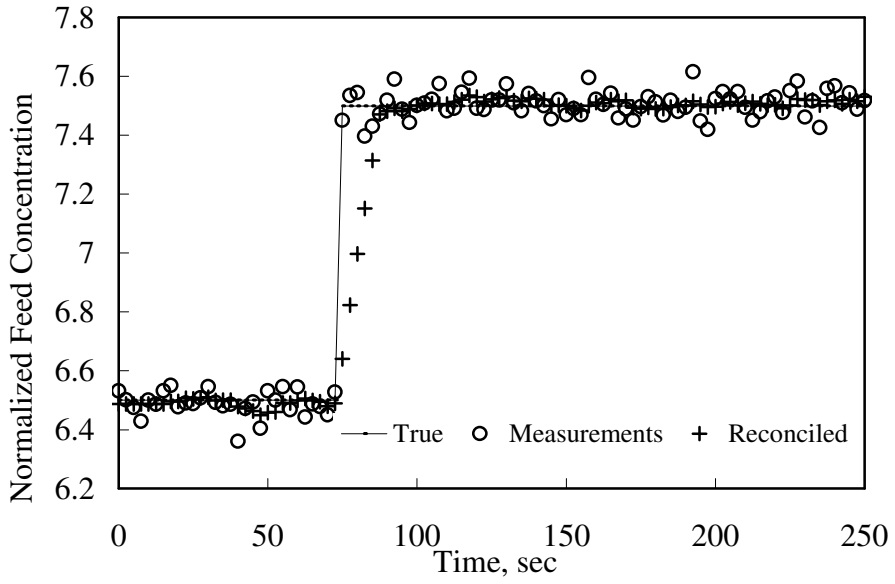


Figure 5-6. Feed Concentration, Liebman *et al*. (1992)

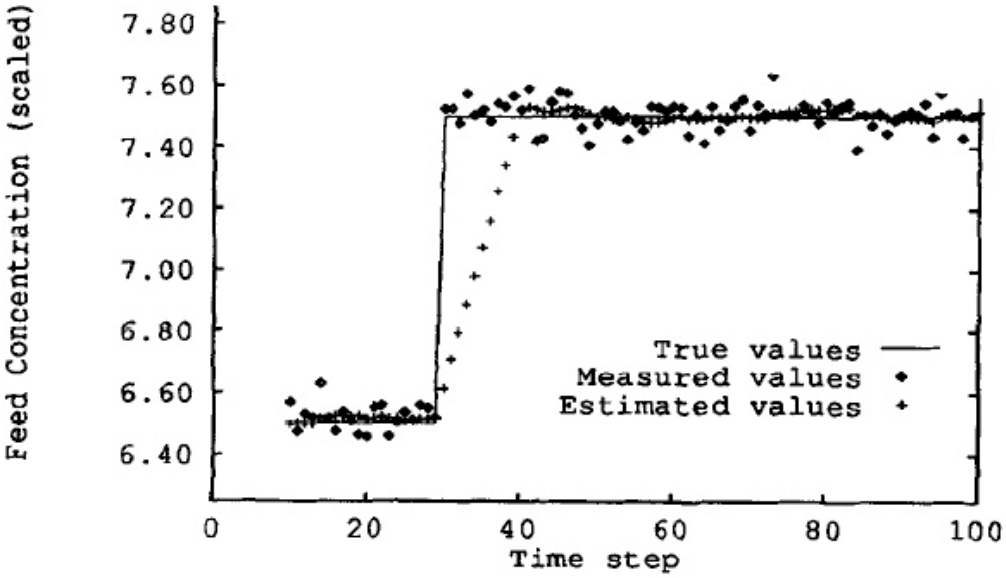


Fig. 7. Step change in feed concentration.

Figure 5-7. Reconciliation of Feed Temperature

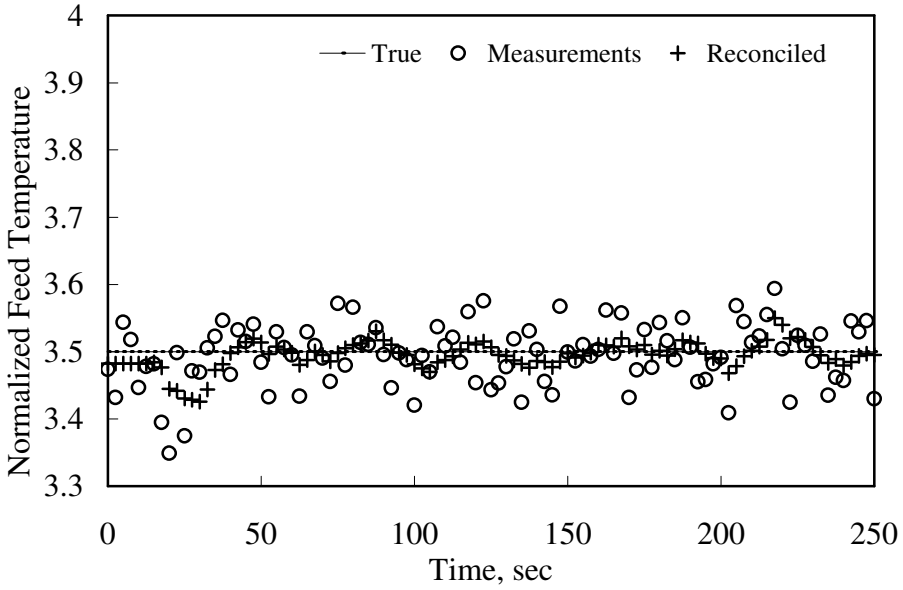


Figure 5-8. Reconciliation of Feed Temperature, Liebman *et al*. (1992)

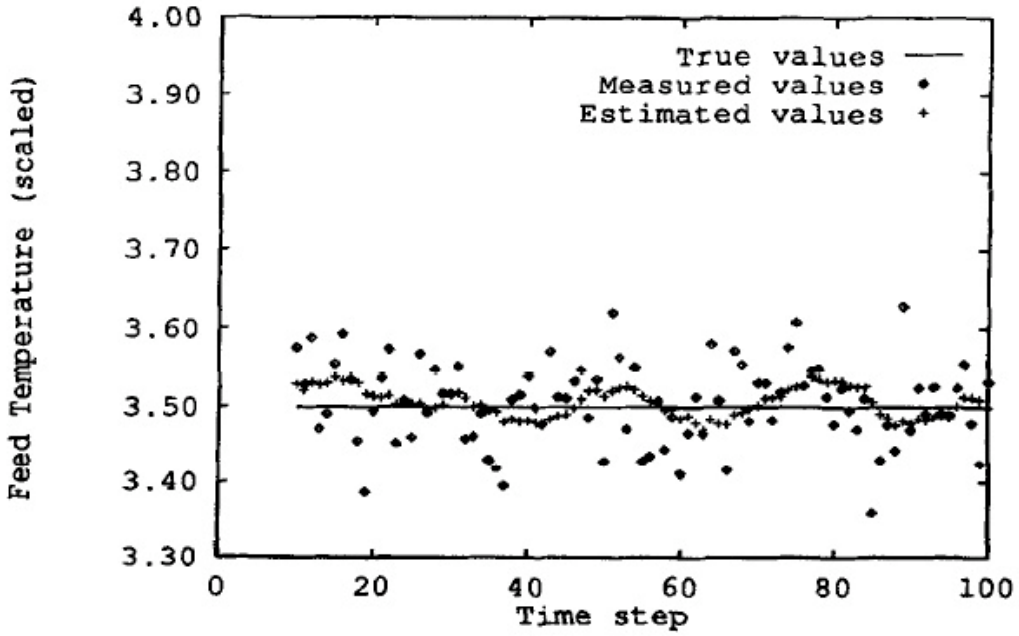


Figure 5-9. Reconciliation of Reactor Temperature

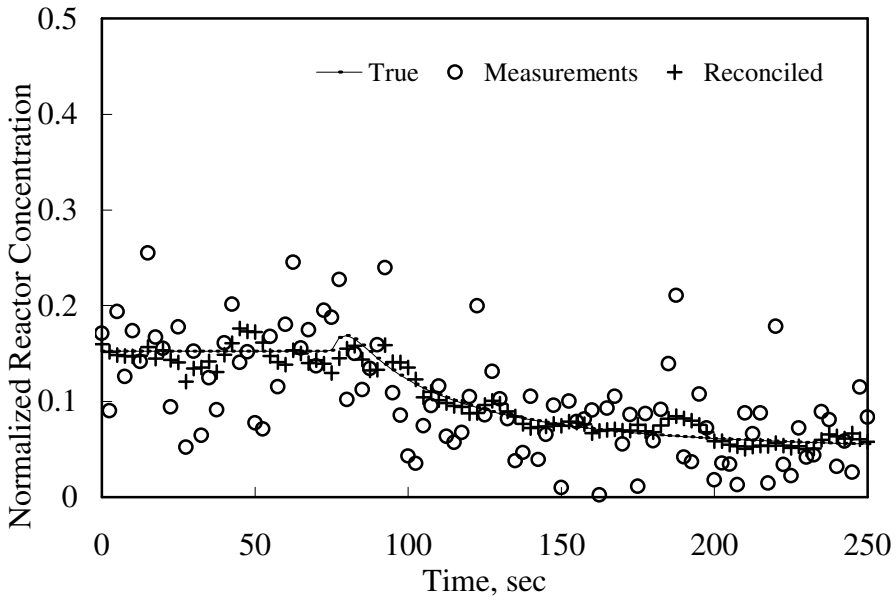


Figure 5-10. Reconciliation of Reactor Concentration, Liebman *et al*. (1992)

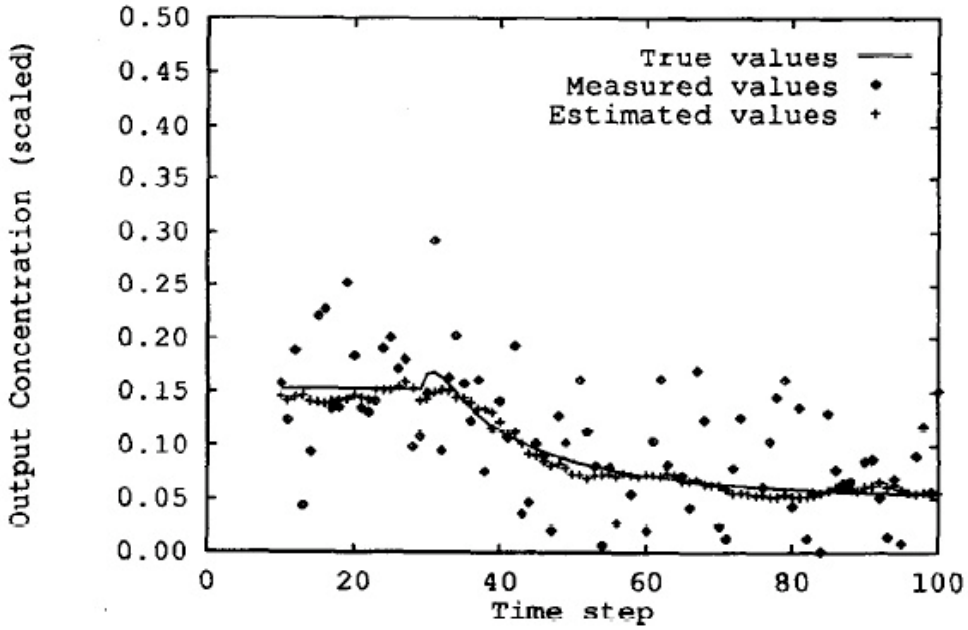


Figure 5-11. Reconciliation of Reactor Temperature

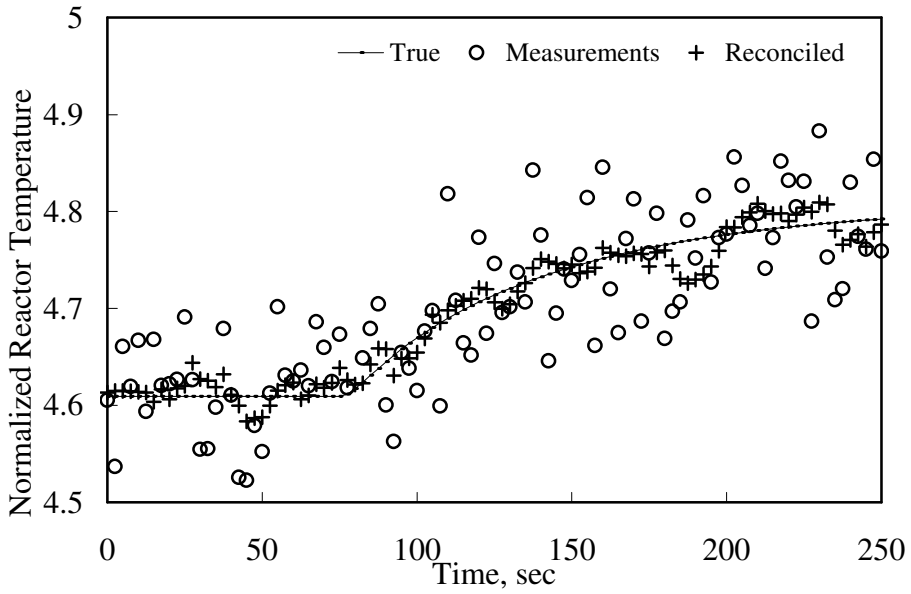
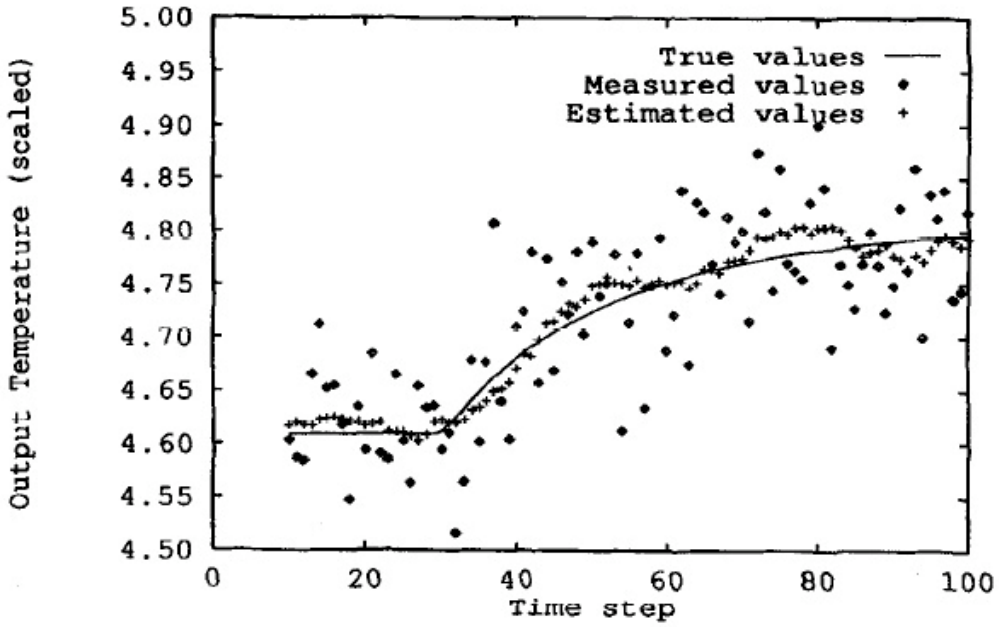


Figure 5-12. Reconciliation of Reactor Temperature, Liebman *et al.* (1992)



5.2.3 Results after Denoising

Denoising has a significant impact on the input to RBR. The above study was repeated with denoising invoked prior to RBR. The denoised measurements were used instead of the raw measurements in the reconciliation objective function. Reconciliation tries to minimize the difference between the model predictions and the denoised measurements. If the denoised measurements are closer to the true process behavior when compared to the raw measurements, then by using the denoised data reconciliation, results will be more accurate.

Measurements from the previously discussed reactor problem were denoised using Daubechie 4 wavelet. This denoising alone removed greater than 50% of the measurement uncertainty (see Table 5-4 and Table 5-5). When the denoised measurements were reconciled, the reconciliation results also improved. This improvement can be seen when comparing the columns 'Plain Reconciliation' and 'Denoising + Reconciliation' in the Table 5-5.

Denoising followed by Rating Based Reconciliation provides results that are equivalent to those from Liebman *et al*. (1992). Results from denoising alone provide better accuracy when compared to the model identification based method proposed by Alici and Edgar (2002). Denoising improved the reconciliation results particularly in reactor concentration and temperature measurements.

Figure 5-13 and Figure 5-14 show the trends in the true, measured, denoised and reconciled data. While measured data are scattered, denoised data are tighter and have an inherent trend. The reconciled data have the least scatter and are closest to the true data.

Table 5-4. Reconciliation Results After Using Denoising

	Measurements		Denoised		Reconciliation	
Variable	Bias	Std. Dev	Bias	Std. Dev	Bias	Std. Dev
A	-0.002	0.047	-0.002	0.024	-0.001	0.009
T	0.000	0.054	0.000	0.024	0.000	0.014
A0	0.002	0.045	0.002	0.065	-0.024	0.125
T0	-0.005	0.046	-0.005	0.025	-0.006	0.020

Table 5-5. Comparison of Reconciliation Results After Using Denoising

	Percentage Reduction in Std. Dev				
	RBR	Denoising	Denoising + RBR	Liebman *et al* (1992)	Alici & Edgar (2002)
A	77.4	67.2	85.5	87.8	62.0
T	70.6	66.4	79.3	77.1	54.0
A0	-	-	-	-	-
T0	54.5	54.1	57.7	65.7	44.0

Figure 5-13. Denoising and Reconciliation Results for Reactor Concentration

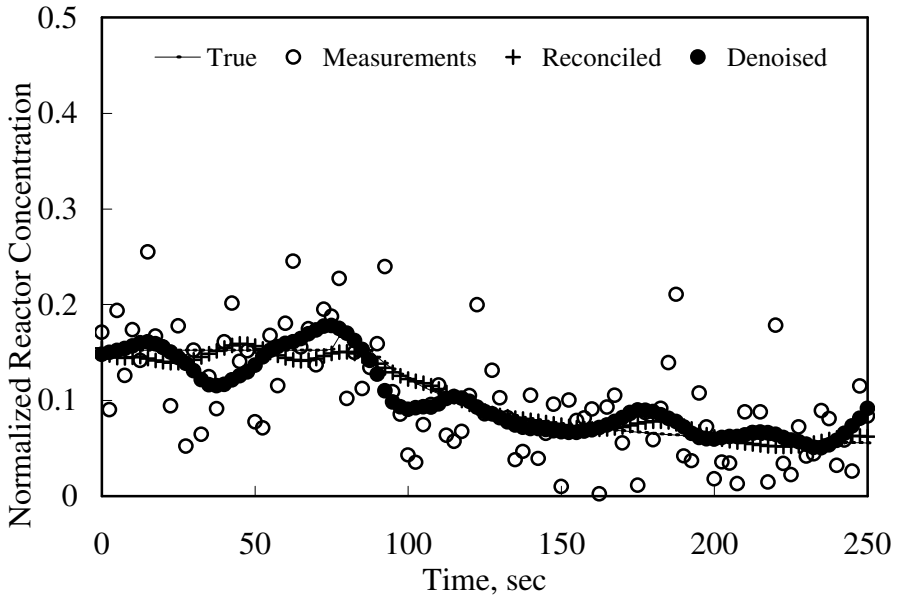
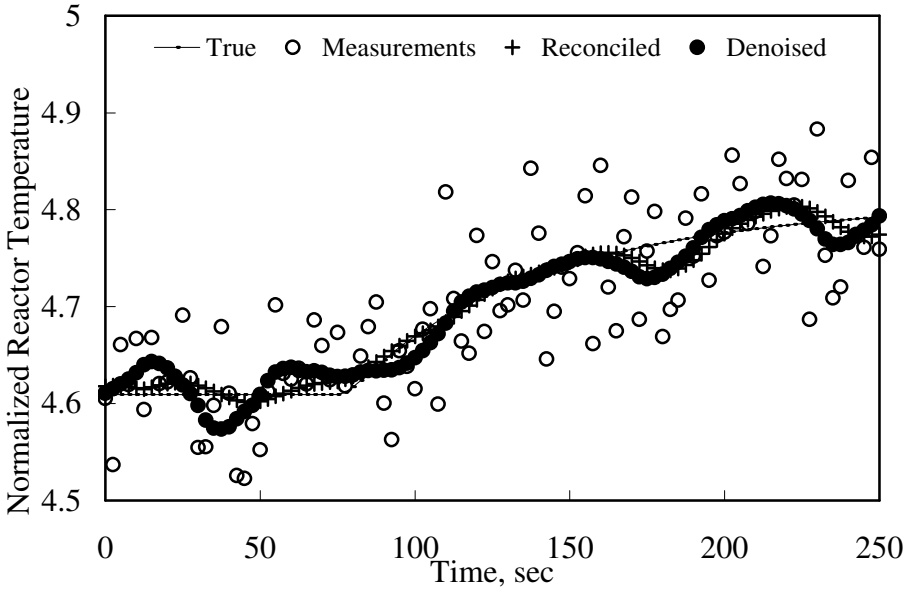


Figure 5-14. Denoising and Reconciliation Results for Reactor Temperature



5.2.4 Size of History Horizon

The history horizon is the amount of past data that is used for reconciliation. It has an effect on the reconciliation results. To study this affect, the previously discussed reactor problem was run using different history horizon values. Results are presented in Table 5-6. Increasing the history horizon improves the reconciliation estimates. However the computation time also increases. History horizon can be treated as a parameter which must be tuned to achieve the necessary improvement in results at an affordable computation cost (Liebman *et al*., 1992).

Table 5-6. Effect of History Horizon on Reconciliation Results

	Percentage Reduction in RMSE	
	History Horizon = last 5 time instances	History Horizon = last 10 time instances
A	77.4	85.4
T	70.6	83.2
A0	-	-
T0	54.5	60.9
Average Computation Time for Reconciliation, seconds	20	57

History horizon forces continuity between measurements separated by time, thus reducing the scatter in the raw measurements. This is similar to smoothing, but the difference is that the process model is also satisfied. The longer the history horizon, the greater the smoothing effect. This is seen by comparing Figure 5-15 and Figure 5-16. The reconciled temperature trend in Figure 5-16 is much smoother and closer to the true behavior when compared to Figure 5-16.

Figure 5-15. Reconciliation of Reactor Temperature, History Horizon = last 5 time steps

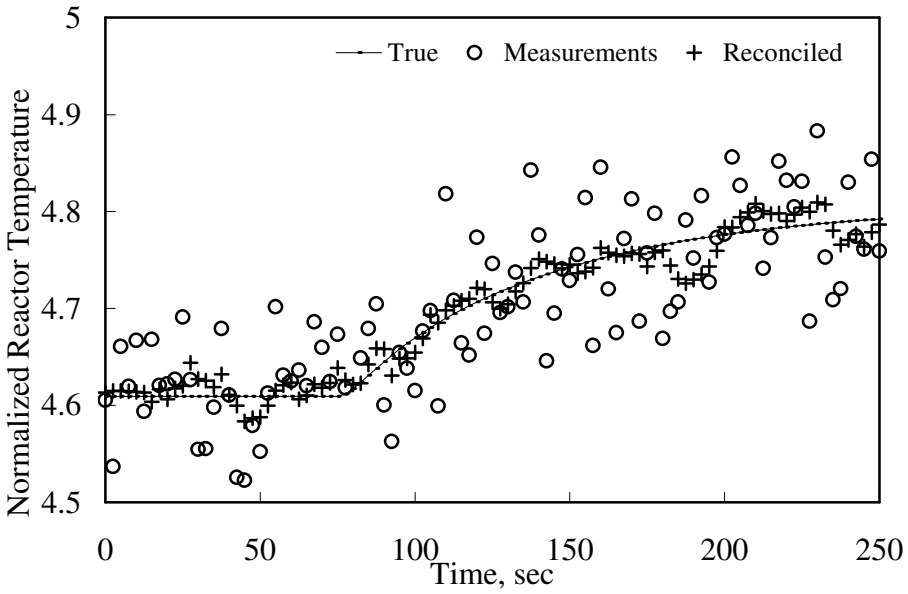
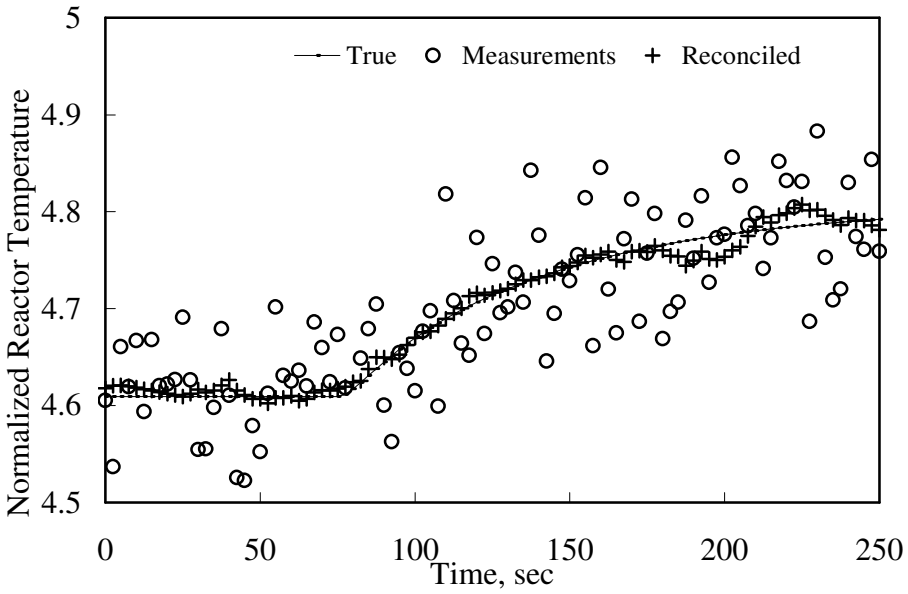


Figure 5-16. Reconciliation of Reactor Temperature, History Horizon = last 10 time steps



5.2.5 Estimation of Unmeasured Variables

Variables such as tray efficiencies and heat transfer coefficients can be estimated by substituting measurements in the process model equations. However, this approach can be problematic when there are redundant measurements and there is noise in the data. Data reconciliation methods can be used under such conditions.

An unmeasured variable estimation problem was devised using the reactor case study. Catalyst activity was included in the model (see equations V-3 and V-4). In the data reconciliation discussed so far, this variable was set to one. Catalyst activity will have a value of one when the extent of reaction is only influenced by reactant concentration and residence time. For this study, data were generated for a situation where catalyst activity gradually reduces by 40%. In order to estimate the activity from the measurements, it is included as one of the optimization variables. However it is not explicitly present in the objective function. The RBR method will find a value for the catalyst activity such that it minimizes the differences between model predictions and the measurements.

The results from Table 5-7 show that the error in the catalyst activity estimation is affected by the size of the history horizon. Increasing the history horizon reduces the error. Using longer history horizon smooths the trends in the measurements and improves their accuracy. Since parameters are estimated from the smoothened measurements, their accuracy also improves. Figure 5-17 and Figure 5-18 show the parameter estimation using different history horizon lengths. The longer history horizon has less scatter and is closer to the true trend.

Table 5-7. Unmeasured Variable Estimation

	Standard Deviation	Bias	RMSE
RBR (hh = 5 time steps)	0.11	-0.06	0.13
RBR (hh = 10 time steps)	0.07	-0.04	0.08
Denoising + RBR (hh = 5 time steps)	0.07	-0.05	0.09
Denoising + RBR (hh = 10 time steps)	0.05	0.05	0.07

Wavelet denoising improves parameter estimation (Table 5-7). It was shown earlier that denoising increased the accuracy of reconciled measurements. Consequently the accuracy of the parameters estimated from them also increases. This effect is stronger when using shorter history horizon. Results in Table 5-7 show that denoising improves the parameter estimation better when using history horizon of 5 steps.

Wavelet denoising produces effect similar to using longer history horizon. Results in Table 5-7 show that using denoising with history horizon of 5 time steps equals to using history horizon of 10 time steps. Comparing Figure 5-17 and Figure 5-19, denoising results in smooth parameter trends. The same effect is seen when increasing the length of the history horizon.

Figure 5-17. Catalyst Activity Estimation (History Horizon = last 5 time steps)

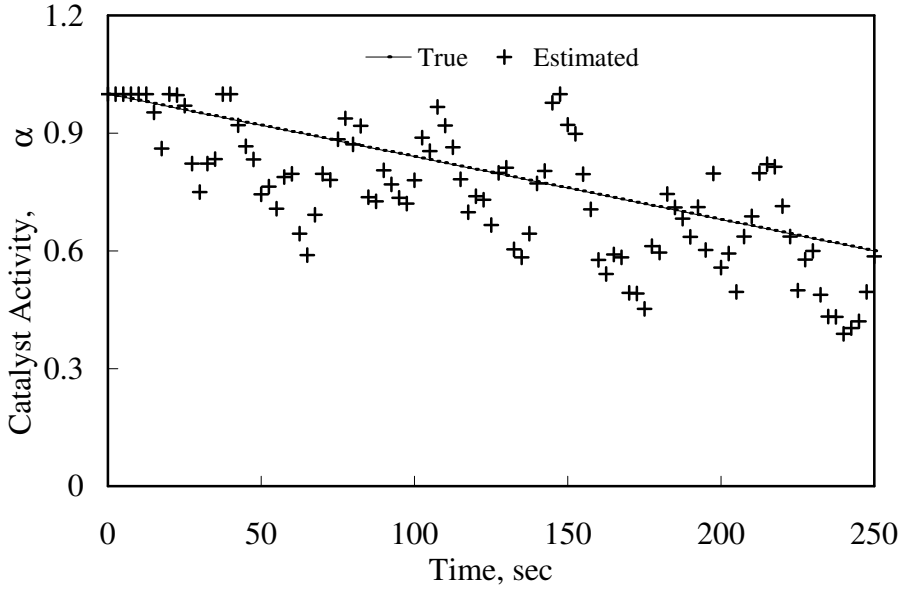


Figure 5-18. Catalyst Activity Estimation (History Horizon = last 10 time steps)

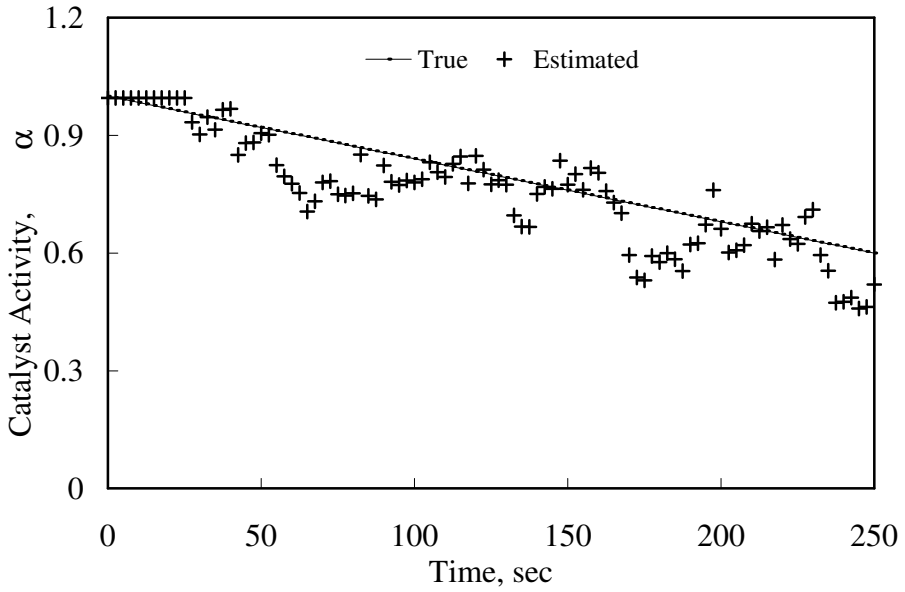
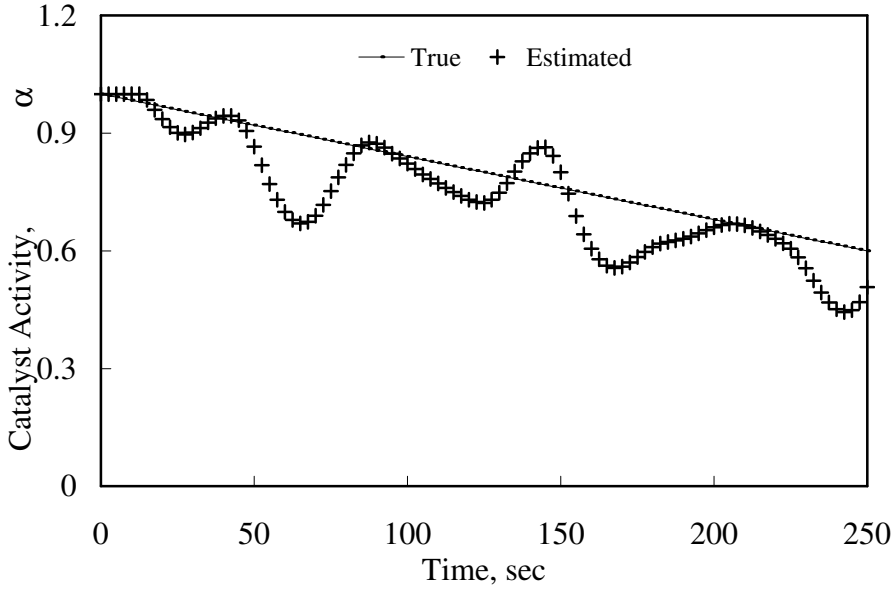


Figure 5-19. Catalyst Activity Estimation After Denoising (History Horizon = 5 time steps)



5.2.6 Computation Time

The RBR method requires longer computation times to solve the reconciliation problem when compared to that required by Liebman *et al*. (1992). See Table 5-8. This is expected because:

1. The process simulation is integrated several times during each iteration of the optimization problem.
2. Time is spent in communication between the simulator and the optimization software.

Table 5-8. Computation Time

History Horizon	RBR		Liebman *et al*. (1992)	
	Average, seconds	Max, seconds	Average, seconds	Max, seconds
5 time steps	20	34	1.14	6
10 time steps	56	115	3.02	16

The RBR simulations were run on a MacBook (2006 Model, 1.83 GHz Intel Core Duo) using virtualization software (1 GB Virtual RAM). Liebman *et al*.(1992) used VAXStation 3200. They do not state the memory and processor information for this machine.

Literature methods that rely on solving the model equations using orthogonal collocation result in large optimization problems (Alici and Edgar, 2002). For complex processes with multiple units the reconciliation problem can consist of thousands of variables. Albuquerque and Biegler (1995) state that methods based on model integration scale significantly better than those relying on orthogonal collocation. As a result the difference in computation speed between the two methods will reduce as the model complexity grows.

5.3 CPD Dimerization Reactor

This case study applies Rating Based Reconciliation methodology to the measurements from a Cyclopentadiene (CPD) Dimerization reactor. The process is much more complex than the previously studied reactor. The model accounts for the changes in the physical properties with the process conditions (i.e. T, P & *x*) and complex phase equilibria relations.

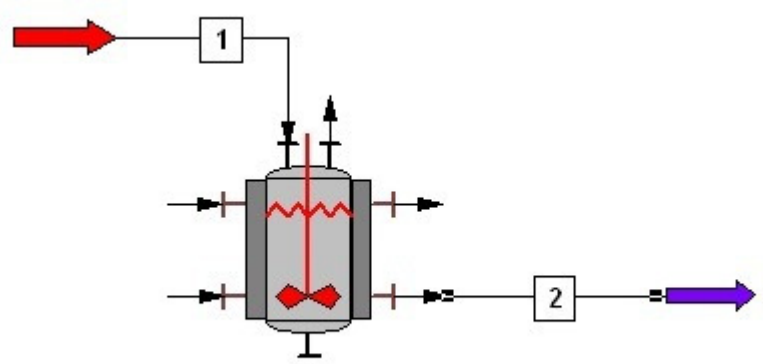
5.3.1 Background

CPD (Cyclopentadiene) dimerization reactor is used in Isoprene purification (Howat, 2004). Isoprene is the monomer pre-cursor to the synthetic equivalent of natural rubber. CPD and isoprene are difficult to separate via distillation. Therefore, CPD is dimerized to form DCPD (DiCycloPentaDiene) in the Dimerization reactor. DCPD can be easily separated from Isoprene and also has value as a product in itself. The Dimerization reaction is auto-catalytic and highly exothermic. Side reactions also occur leading to the formation of co-dimers, oligomers and polymers. These unwanted reactions are exothermic as well. Therefore improperly controlled CPD Dimerization reaction can lead to runaway conditions. Howat (2004) compiled the calorimetric and thermodynamic data for the dimerization, oligomerization and polymerization reactions. He built a CHEMCAD simulation of the process to analyze the excursion potential of these reactions.

The CHEMCAD simulation used in this case study (Figure 5-20) was built from the data compiled by Howat (2004). It consists of an adiabatic reactor that can run in dynamic

mode including the holdup volume in simulations. The feed consists of Isoprene (IPM), CPD, and n-Pentane (nC5). CPD dimerizes in the reactor to form DCPD. The measurements consist of the inlet, reactor and exit stream conditions (i.e. Temperature, T; Pressure, P and Component molar flows).

Figure 5-20. CHEMCAD Simulation of Adiabatic CPD Dimerization Reactor



5.3.2 Reconciliation Results

The process situation studied here consists of changes in the reactor feed temperature and their affect on the reaction mixture and the exit streams. Figure 5-21 and Figure 5-22 show these changes in the feed temperature and their corresponding impact on the reactor temperature and contents. The feed temperature undergoes a short lived 5 F spike at the 5^{th} minute. It later reduces by 5 F for few minutes starting at the 10^{th} minute. The reactor temperature drops by 5 F with the feed temperature reduction. With lowered reactor temperature the rate of CPD dimerization reduces slightly. As a result the CPD content in the reactor increases by 5%.

Twenty five minutes of the process operation were studied. Measurements were generated twice each minute. They were denoised using Daubechie 4 wavelet and then reconciled using the RBR method. The reconciliation program was written in MATLAB. It controlled the CHEMCAD simulations through a Microsoft Excel file. History horizon of 2.5 minutes was used (equal to five time steps). The results from this reconciliation are given in Table 5-9 and Table 5-10. While Table 5-9 lists the standard deviation and bias, Table 5-10 provides a more comprehensive look at the results.

Table 5-10 provides a statistic called success rate, %. This is the fraction of total runs when the reconciled (or denoised) measurements were closer to the true values than the raw measurements. The success rate must be greater than 50% in order for the reconciliation method to be called successful in improving the estimates for a given measurement. MacDonald and Howat (1988) are the only researchers who report this statistic. They report an average 65% success rate for their steady state reconciliation methods.

Results in Table 5-10 show that for all of the measurements the RMSE reduced by at least 50%. The success rate for the measurements is around 70 to 90%. This means that most of the time the reconciliation results are closer to the true behavior than the measurements. This is better than that reported by MacDonald and Howat (1988).

Figure 5-21. CPD Dimerization Reactor Transient Conditions, Reactor Temperature.

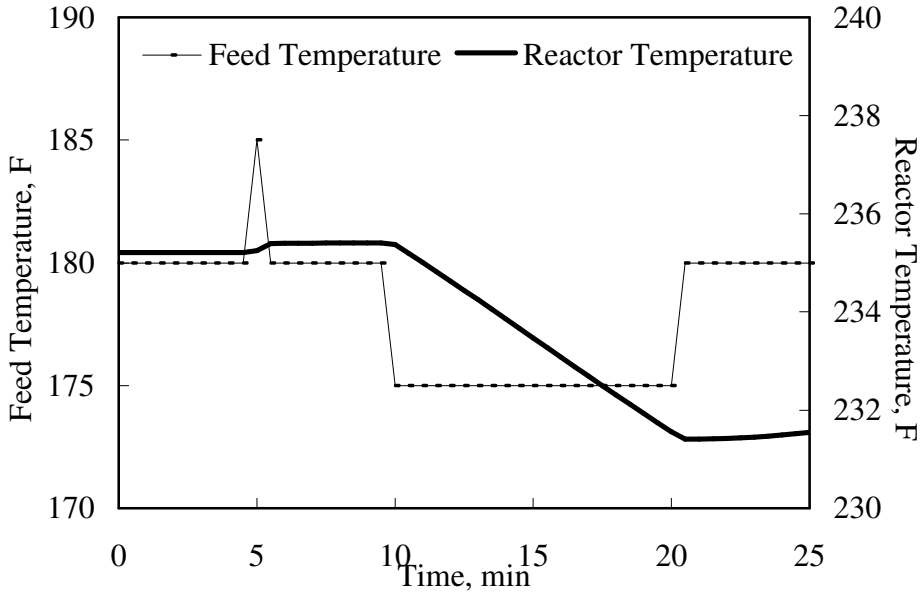


Figure 5-22. CPD Dimerization Reactor Transient Conditions, CPD in Reactor.

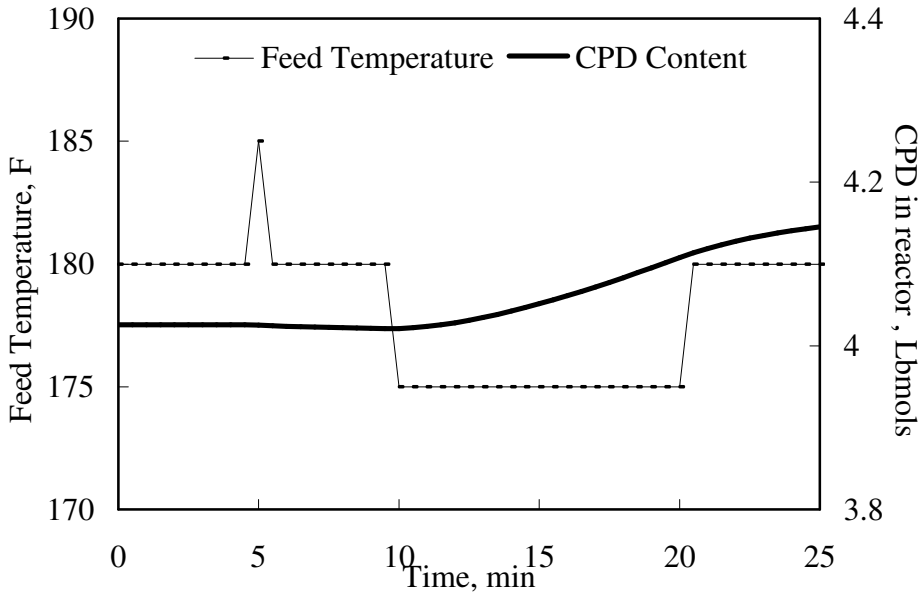


Table 5-9. Detailed Reconciliation Results for the CPD Dimerization Reactor

	Std. Dev			Bias		
	Measured	Denoised	Reconciled	Measured	Denoised	Reconciled
T_1, F	4.70	3.08	2.49	-0.56	-0.62	0.00
P_1, Psia	0.22	0.10	0.06	-0.03	-0.04	-0.01
IPM_1, lbmol/hr	0.89	0.64	0.39	-0.07	-0.08	-0.02
CPD_1	0.91	0.34	0.26	0.02	0.02	0.03
$nC5_1$	0.56	0.23	0.13	0.08	0.08	0.08
$DCPD_1$	0.00	0.00	0.00	0.00	0.00	0.00
T_2, F	4.59	2.51	0.20	0.43	0.40	0.06
P_2, Psia	0.22	0.09	0.06	0.00	0.00	-0.01
IPM_2	1.05	0.55	0.22	0.25	0.25	-0.03
CPD_2	1.07	0.56	0.19	-0.02	-0.02	0.04
$nC5_2$	0.48	0.17	0.08	0.08	0.07	0.08
$DCPD_2$	0.53	0.24	0.06	-0.01	-0.02	-0.02
$T_{Reactor}$, F	0.43	0.17	0.20	0.05	0.05	0.06
$P_{Reactor}$, Psia	0.20	0.12	0.06	0.03	0.03	-0.01
$IPM_{Reactor}$	0.09	0.04	0.02	-0.03	-0.03	-0.01
$CPD_{Reactor}$	0.10	0.04	0.02	0.01	0.01	0.01
$nC5_{Reactor}$	0.05	0.03	0.01	0.01	0.01	0.01
$DCPD_{Reactor}$	0.05	0.02	0.01	-0.01	-0.01	-0.01

Table 5-10. RMSE Reduction and Success Rate, Dimerization Reactor

	Reduction in RMSE, %		Success Rate, %	
	Denoising	Denoising + RBR	Denoising	Denoising + RBR
T_1, F	33.6	47.5	68.6	68.6
P_1, Psia	52.4	73.9	74.5	88.2
IPM_1, lbmol/hr	27.5	55.6	58.8	68.6
CPD_1	62.7	71.8	70.6	74.5
$nC5_1$	57.0	72.8	82.4	84.3
T_2, F	44.9	88.8	72.5	96.1
P_2, Psia	61.0	73.8	80.4	86.3
IPM_2	44.3	79.6	70.6	88.2
CPD_2	48.1	81.8	80.4	88.2
$nC5_2$	61.6	76.3	78.4	86.3
$DCPD_2$	53.5	88.0	68.6	96.1
$T_{Reactor}$, F	59.2	52.8	78.4	84.3
$P_{Reactor}$, Psia	39.2	69.4	68.6	76.5
$IPM_{Reactor}$	48.7	74.4	62.7	76.5
$CPD_{Reactor}$	56.6	80.4	74.5	90.2
$nC5_{Reactor}$	47.8	63.7	72.5	82.4
$DCPD_{Reactor}$	57.3	78.6	80.4	92.2

Figures 5-23 through 5-26 show detailed trends from reconciliation in feed and reactor temperatures and component flow rates. Figure 5-23 shows the reconciliation of feed temperature. At t = 5 minutes, there is a short bump in the feed temperature. The temperature also drops by 5 F at the 10^{th} minute and reverts back at the 20^{th} minute. These changes were placed to see how reconciliation and denoising algorithms process them. The graphs show that the algorithms diffuse the short spike and smooth the sharp changes in the feed temperature. The resulting trend contains 70% less error than the measurements.

The reconciled reactor temperatures match the true behavior (Figure 5-24). More than 50% of the error in the measurements is reduced after reconciliation. The reconciled temperatures are better than the measurements 85% of the time.

Reconciliation removed most of the error present in the CPD reactor content measurements (Figure 5-25). Denoising results contain a wavy trend to them while the true process behavior does not. In spite of this, denoising results contain 50% less error when compared to the measurements. Reconciliation after denoising reduces the error by 80%. Reconciliation results have better accuracy than the measurements 90% of the time.

The DCPD flow rate measurements improve substantially after reconciliation (Figure 5-26). The error reduces by 50% and 90% after denoising and reconciliation respectively. Reconciliation results are better than measurements 95% of the time. The error reduction reported here is on par with those reported by Liebman *et al*. (1992), MacDonald and Howat (1988), Alici and Edgar (2002) and Vachhani *et al* (2005).

Figure 5-23. Reconciliation of Feed Temperature Measurement

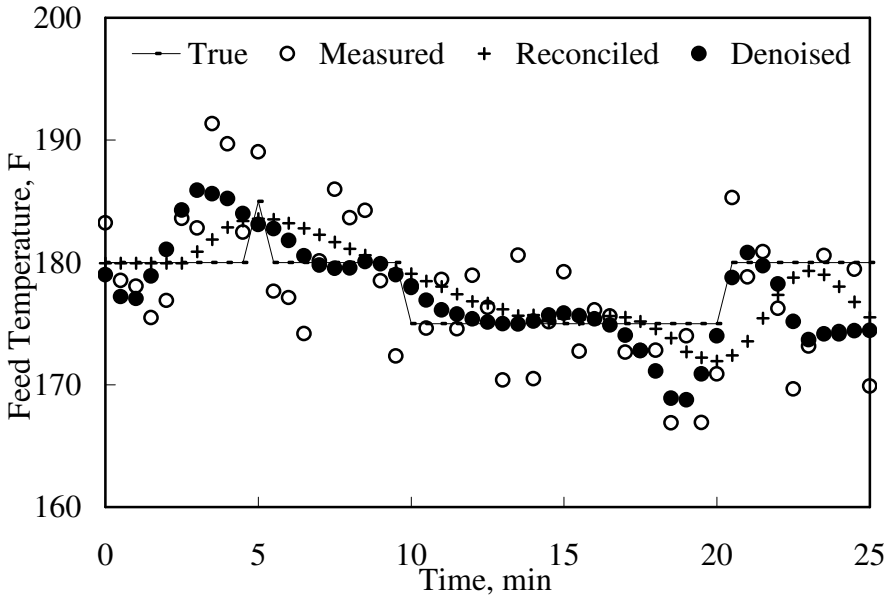


Figure 5-24. Reconciliation of Reactor Temperature Measurement

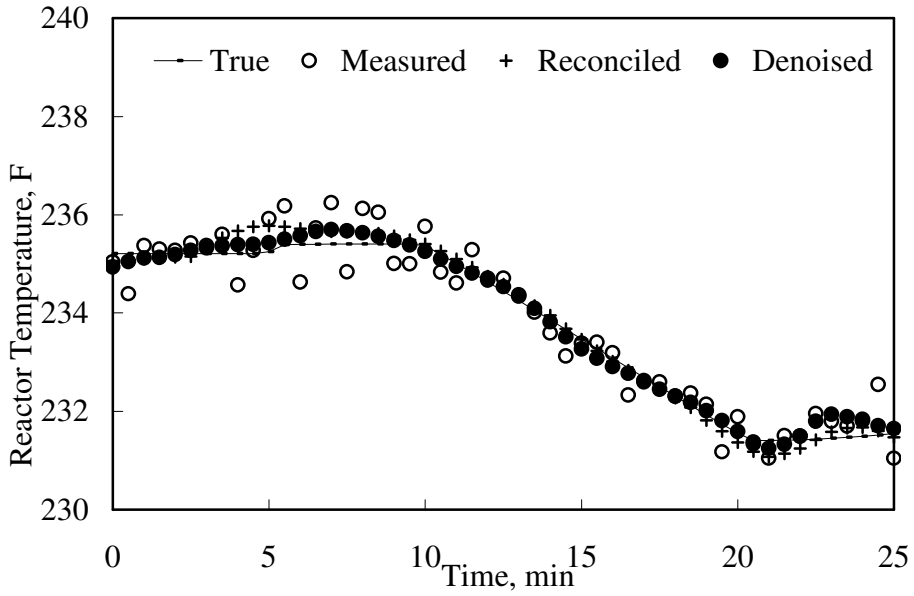


Figure 5-25. Reconciliation of CPD in Reactor Measurement.

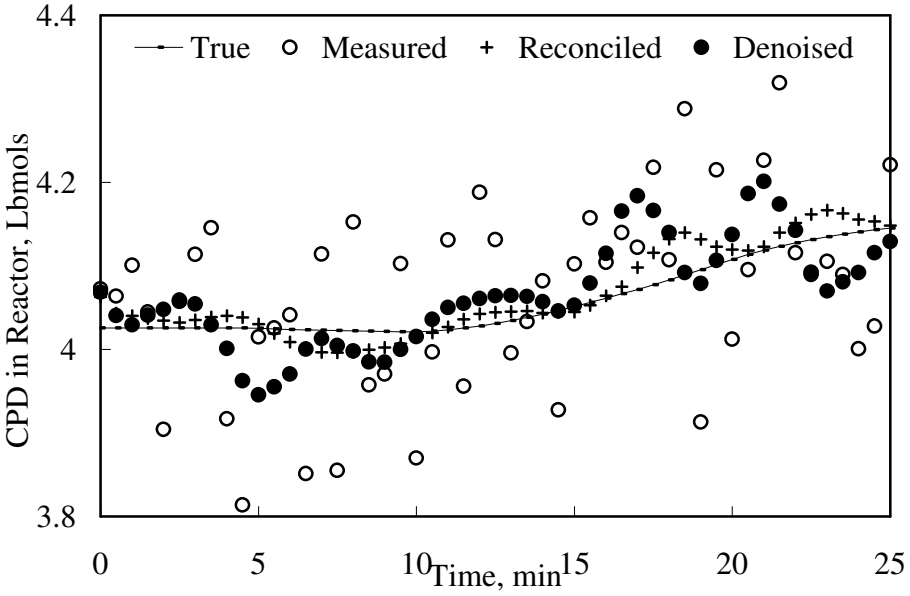
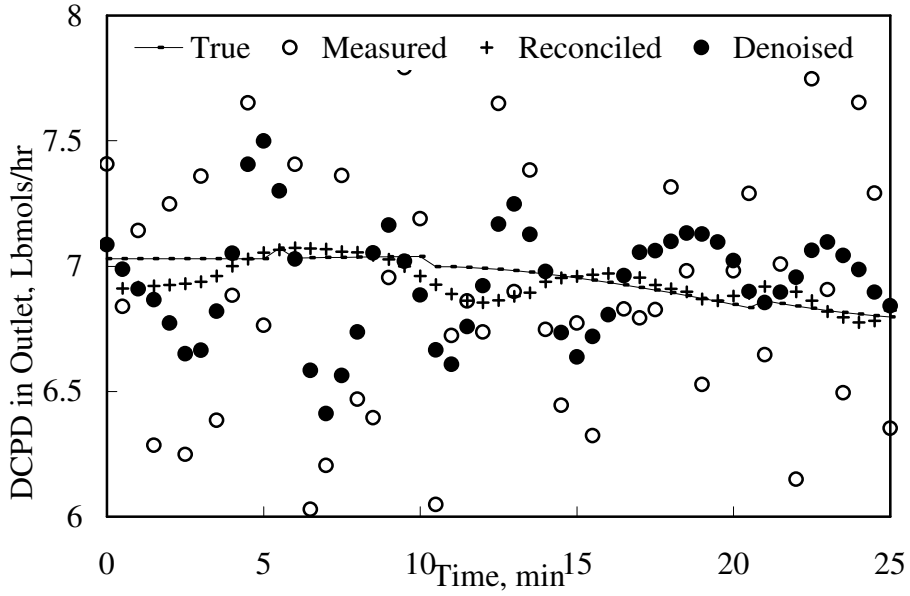


Figure 5-26. Reconciliation of DCPD Flow in Exit Stream.



5.3.3 Feed Temperature Estimation

Process measurements are often incomplete. In those cases, an effective analysis method must be capable of estimating values for the missing measurements. The CPD reactor was reanalyzed with the feed temperature missing from the set of measurements. It is estimated by the RBR method during the reconciliation of the available measurements. Transient behavior is introduced into the reactor operation through the gradual reduction of feed temperature (Figure 5-27). This reduces the reactor temperature. Measurements were generated for this process behavior and reconciled using the RBR method. Denoising was not applied to the data. This is to measure the performance of the RBR method in the absence of denoising.

Figure 5-27. CPD Dimerization Reactor Transient Conditions, Feed and Reactor Temperatures.

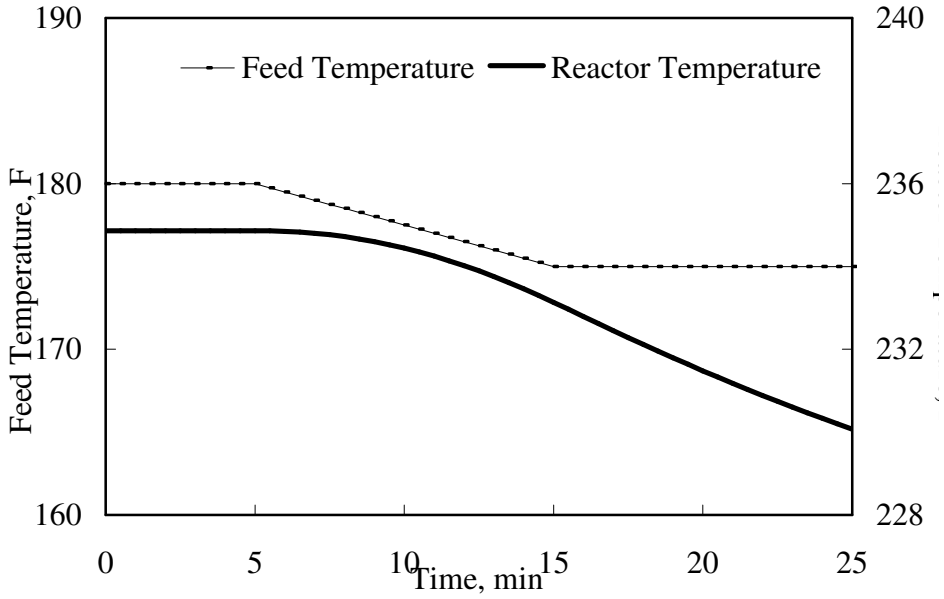


Figure 5-28 shows the feed temperature estimated from the available measurements and compares it with the true data. The first row in Table 5-11 shows the statistics from the estimation. There is no bias in the estimated feed temperature. The estimation traces the true feed temperature with scatter. The standard deviation of the scatter is 3.3 F, 1.7% of the measurement value. It is difficult to determine whether the trend in the estimated temperatures match the 5 F drop in the true feed temperature. This is because a change of 5 F is not much larger than the standard deviation of the estimation. However, it can be said that the RBR methodology estimated feed temperature successfully with less than 2% error.

Figure 5-28. Feed Temperature Estimation.

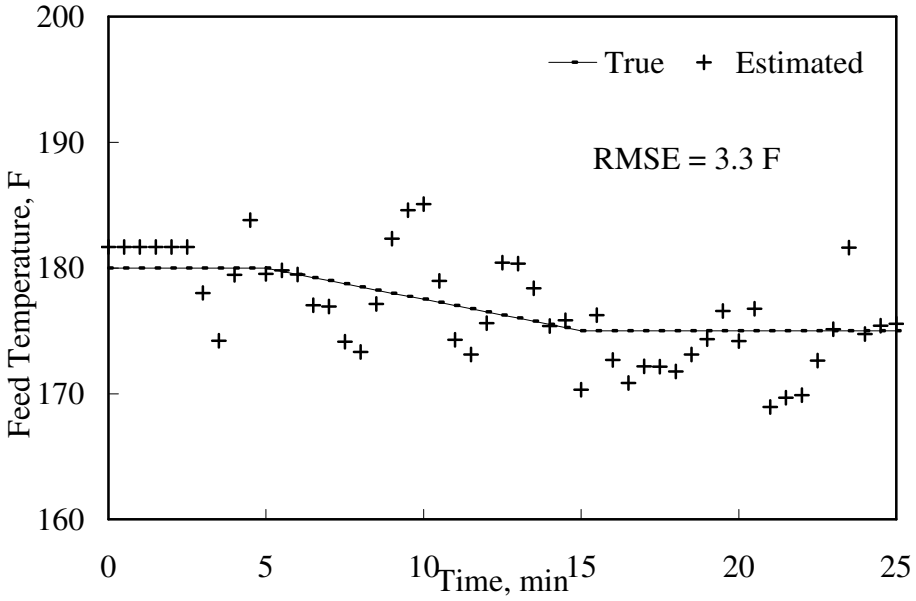


Table 5-11 and Table 5-12 show the performance of RBR in reconciling the available measurements. These results do not use wavelet denoising. Most of the measurements

undergo greater than 60% reduction in the error. The reconciliation success rate is also greater than 80% for a majority of the measurements. The absence of feed temperature, a key measurement, did not hamper the error reduction seen in the results.

Table 5-11. Reconciliation Results, Feed Temperature Estimation

	Std. Dev		Bias	
	Measured	Reconciled	Measured	Reconciled
T_1, F	-	3.28	-	-0.3
P_1, Psia	0.19	0.06	-0.04	-0.04
IPM_1, lbmol/hr	1.18	0.32	0.06	-0.01
CPD_1	1.08	0.43	0.10	0.07
$nC5_1$	0.53	0.21	0.08	0.07
$DCPD_1$	0.00	0.00	0.00	0.00
T_2, F	5.04	1.61	1.02	0.21
P_2, Psia	0.18	0.06	-0.03	-0.04
IPM_2	0.89	0.31	-0.06	0.06
CPD_2	1.03	0.20	0.01	0.01
$nC5_2$	0.49	0.14	0.09	0.04
$DCPD_2$	0.43	0.09	-0.07	-0.04
$T_{Reactor}$, F	0.52	0.31	0.05	-0.02
$P_{Reactor}$, Psia	0.22	0.06	-0.04	-0.04
$IPM_{Reactor}$	0.11	0.03	0.01	0.01
$CPD_{Reactor}$	0.09	0.02	-0.02	-0.01
$nC5_{Reactor}$	0.05	0.01	0.00	0.00
$DCPD_{Reactor}$	0.06	0.01	-0.01	-0.01

Table 5-12. RMSE Reduction and Success Rate,

	Reduction in RMSE, %	Success Rate, %
T_1, F	-	-
P_1, Psia	63.0	80.4
IPM_1, lbmol/hr	72.7	90.2
CPD_1	60.2	78.4
$nC5_1$	58.1	88.2
T_2, F	68.4	92.2
P_2, Psia	58.2	72.5
IPM_2	64.6	80.4
CPD_2	80.4	84.3
$nC5_2$	70.5	86.3
$DCPD_2$	76.7	82.4
$T_{Reactor}$, F	40.3	68.6
$P_{Reactor}$, Psia	67.4	86.3
$IPM_{Reactor}$	74.1	86.3
$CPD_{Reactor}$	74.6	90.2
$nC5_{Reactor}$	73.3	84.3
$DCPD_{Reactor}$	75.0	92.2

The RBR method was successful in reconciling the measurements from a CPD dimerization reactor. It improved the accuracy of measurements with sharp changes (e.g. feed temperature) and those with gradual changes (e.g. CPD content in the reactor). The method was able to simultaneously estimate unmeasured feed temperature accurately and reduce the measurement error.

5.4 Batch Distillation

Batch distillation, a process with depleting inventory; changing compositions and temperatures; and, multiple stages, is the rigorous test for the Rating Based Reconciliation (RBR).

5.4.1 Simulating Batch Distillation

The example separates C_3 to C_6 alkanes. It is built using the batch distillation example provided in CHEMCAD documentation (Chemstations, 2010). The reboiler is fed with a mixture of C_3 to C_6 alkanes. The column is initially dry. Composition and temperature profiles are established in the column by running it at a total reflux condition. In total reflux condition, the reboiler contents are heated. The generated vapor passes through the column and condenses in the condenser. The condensate is completely recycled to the column. By running in this fashion for few minutes, the temperature and composition profiles in the column reach a steady state. At this point the product withdrawal is started. Measurements are generated from this point on. They are reconciled using the RBR methodology.

The batch distillation module in Chemcad does not allow interaction with Excel. An alternative was required to perform reconciliation. The alternative setup consists of dynamic reactor and a multi-stage column (Figure 5-29). The dynamic reactor without reaction and impeller functions as the reboiler. The multi-stage column is the column portion of the batch distillation. The condenser is a part of the column. This arrangement was found to match very closely to that of the default batch distillation (see Section 9).

The following are required to build the alternative version of batch distillation simulation:

1. Thermodynamics (Table 5-13).
2. Equipment Specifications (Table 5-13).
3. Operating Specifications (Table 5-14 & Table 5-15).
4. Initial Reboiler Conditions (Table 5-14).

Figure 5-29. Alternative Batch Distillation Arrangement

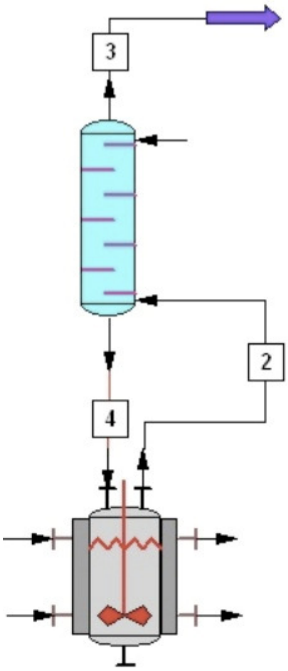


Table 5-13. Thermodynamics and Equipment Specifications.

Thermodynamics		*Reactor (i.e. Reboiler)*	
K-value Model	PR	Reactor Volume	250 ft^3
Enthalpy model	PR	Vessel Diameter	4.8 ft
heat capacity	Polynomial	Wall thickness	0.0833 ft
		Wall density	490.75 lbs/ft^3
Multi-Stage Column		Wall specific heat	0.1125 Btu/lb-F
No. of Stages	3	Wall thermal conductivity	27.5 Btu/hr-ft-F
Condenser Type	Total Condenser	Impeller Dia	1 ft
Pressure Drop per stage	0.1 Psi	Impeller speed	0.00001 hz
Condenser Pressure Drop	2 Psi		

Table 5-14. Specifications for the Reactor Acting as the Reboiler

Reactor Specifications	
Heat Duty	0.8 MMBTU/Hr
Constant Pressure	102.2 Psia
Initial Charge	
Temp, F	184.63
Press, Psia	102.2
C3, lbmols	10
C4	30
C5	10
C6	50

Table 5-15. Multi-Stage Column Operating Specifications

Column Specifications	
Distillate Rate	17.5 Lbmol/Hr
Liquid Holdup	Negligible
Condenser Holdup	Negligible
Vapor Holdup	Negligible
Initial Conditions	Dry Tray Startup

The reboiler is maintained at constant pressure. Since there is no bottoms product, outgoing vapor flow (stream # 2) is modulated to keep the pressure constant. The multi-stage column is run with constant pressure drop. There is no holdup associated with the stages and the condenser.

A different simulation is used to establish the total reflux conditions (Figure 5-30). The initial charge to the reboiler is specified in this simulation. The column is initially dry. Composition and temperature profiles are established in the column by heating the reboiler contents and returning the distillate completely to the column. Condenser heat duty is set equal to the reboiler heat duty. This is continued untill the composition and temperature profiles reach a steady state. The simulation is then changed such that the distillate stream becomes the product stream (Figure 5-29). The steady state column profiles from the total reflux simulation become the initial conditions (Table 5-16). The condenser is then operated such that a specified amount of distillate is made. The resulting simulation with column profiles is used for representing the batch distillation where product is withdrawn.

Figure 5-30. Total Reflux Conditions

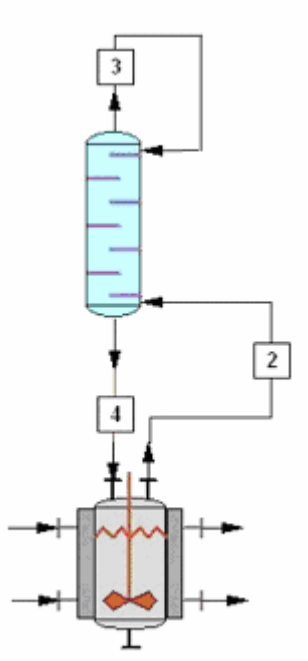


Table 5-16. Initial Conditions for the Batch Distillation Simulation

	Reboiler	Stream # 2	Stream # 3	Stream # 4
Temp, F	185.36	185.36	62.20	112.76
Press, Psia	102.2	102.2	100	102.1
C3, Lbmols/hr	10.00 lbmols	5.12	0.09	5.14
C4	30.00 lbmols	6.37	0.01	6.37
C5	10.00 lbmols	0.98	0.00	0.99
C6	50.00 lbmols	2.09	0	2.09
Heat to Reboiler, MMBTu/Hr	0.8			
Distillate Rate	17.5 lbmols/hr			

5.4.2 Data Generation

This scenario is a 36 minute campaign with the majority of propane collected overhead. Measurements for this operation (i.e. temperatures, pressures and molar flows) are generated by adding normally distributed random error to true data (see Table 5-17). Normally distributed random error is representative of error found in measurements. True data were generated by running the batch distillation simulation (Figure 5-29) using the initial conditions given in Table 5-16.

Table 5-17. Measurement Standard Deviations

	Standard Deviation
Temperature	4 F
Pressure	0.2 Psia
C3	1 lbmols
C4	1 lbmols
C5	0.5 lbmols
C6	0.5 lbmols
Reboiler Duty	0.04 MMBtu/Hr
Distillate Rate	1 lbmol/hr

5.4.3 Results without Denoising

The measurements consist of the temperature, pressure and composition data for the distillate, intermediate streams (i.e. streams 2 & 4 in Figure 5-29) and the reboiler conditions. They are sampled twice every minute. They are reconciled using the RBR methodology with a history horizon of 5 time steps, i.e. 2.5 min. During reconciliation, measurements at the first time step in the calculation horizon are specified as the initial conditions for the batch distillation simulation. The simulation is run for the duration of this history horizon with the distillate rate and the reboiler heat duty set constant at the initial conditions value.

Reconciliation results are more accurate than the raw data at least 60% of the time for most of the measurements (see success rate in Table 5-18 and Table 5-19). This is comparable to that reported by MacDonald and Howat (1988). They report an average success rate of 65% after reconciling steady state flash measurements. Here, the unit operation studied, batch distillation, is more complex than flash.

The temperature, pressure, reboiler and distillate composition measurements see considerable reduction in error ranging from 15% to 90% (see % reduction in RMSE, Table 5-18 and Table 5-19). In the case of very few intermediate stream measurements the error increases after reconciliation (e.g. Butane flow measurement in Streams 2 and 4, see Table 5-19).

Table 5-18. Reconciliation Results for Reboiler and Distillate Measurements

	Reconciliation Success Rate, %	% Red in RMSE
$T_{Reboiler}$, F	64	46
$P_{Reboiler}$, Psia	75	61
$C3_{Reboiler}$, lbmol	74	44
$C4_{Reboiler}$	63	35
$C5_{Reboiler}$	73	31
$C6_{Reboiler}$	64	23
T_3, F	62	15
P_3, Psia	78	57
$C3_3$, lbmols/hr	67	31
$C4_3$	73	28
$C5_3$	96	72
$C6_3$	93	93
Q, MMBtu/hr	62	25
Dist rate, lbmols/hr	68	22

Table 5-19. Reconciliation Results For the Intermediate Stage Measurements.

	Reconciliation Success Rate, %	% Red in RMSE
T_2, F	73	46
P_2, Psia	77	61
$C3_2$, lbmols/hr	49	-27
$C4_2$	42	-46
$C5_2$	66	32
$C6_2$	49	-24
T_4, F	64	20
P_4, Psia	78	64
$C3_4$, lbmols/hr	60	2
$C4_4$	34	-79
$C5_4$	68	43
$C6_4$	49	-35

Reconciled reboiler and distillate measurements have 15% - 60% lower error than the raw measurement data (Table 5-18). This is reflected in Figure 5-31 to Figure 5-38. Pentane and hexane flows see 70% and 90% error reduction respectively after reconciliation. Due to relatively low volatility these compounds will be present in small quantities (Mole fraction of $C_5 < 5\%$, $C_6 < 1\%$) in the distillate. The raw data indicate much higher values for these compounds in the distillate due to measurement error (Figure 5-37 and Figure 5-38). The model could predict their low concentrations. As a result the reconciliation results (i.e. model predictions) have 95% less error than the raw data for these measurements.

There is no error reduction after reconciliation for the following measurements: Butane and Hexane in streams 2 and 4 and Propane in stream 2. Figure 5-39 to Figure 5-41 show the trends in these measurements. Greater than 50% of the time the measurements are more accurate than the reconciliation results. The temporal disconnect between the streams 2 and 4 (Figure 5-29) is the reason. The simulations carried out during the reconciliation are sequential modular in nature. At any given time instant, stream 2 represents the vapor from the reboiler at time *t* and stream 4 (i.e. cut stream) represents the liquid from the column at time *t-1*. Reconciliation does not account for this temporal disconnect among the measurements. Consequently the error increases for some of components in these two intermediate streams. This error would be minimized if an equation based solver is used for simulating multiple units. While using a sequential modular simulator, this error may be reduced by moving the measurements for the cut streams by one time step into the past.

Figure 5-31. Reconciliation of Reboiler Temperature

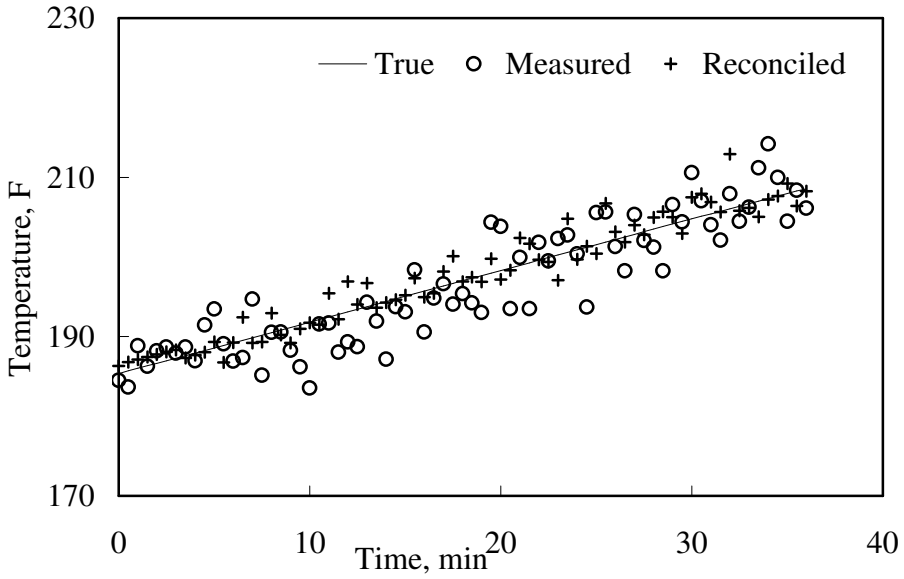


Figure 5-32. Reconciliation of Reboiler Pressure

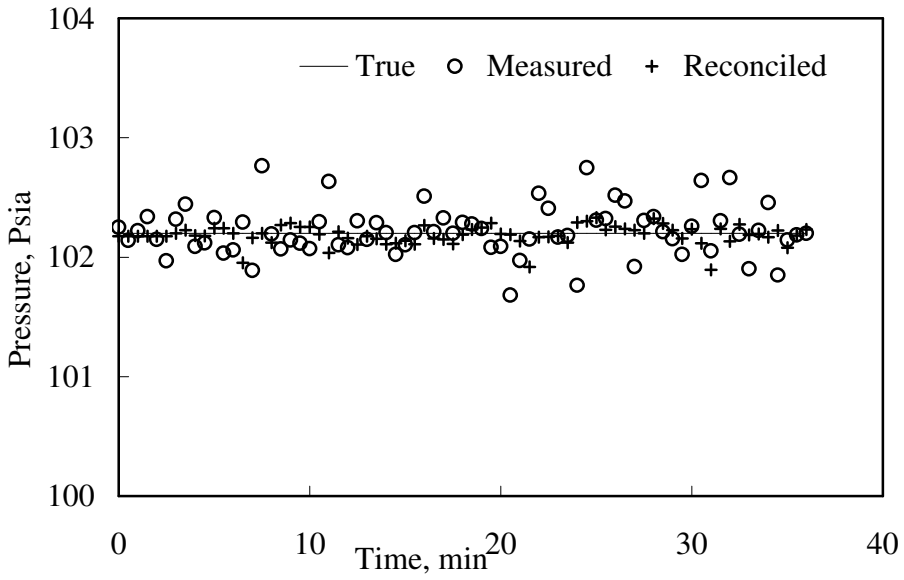


Figure 5-33. Reconciliation of Propane Content in the Reboiler.

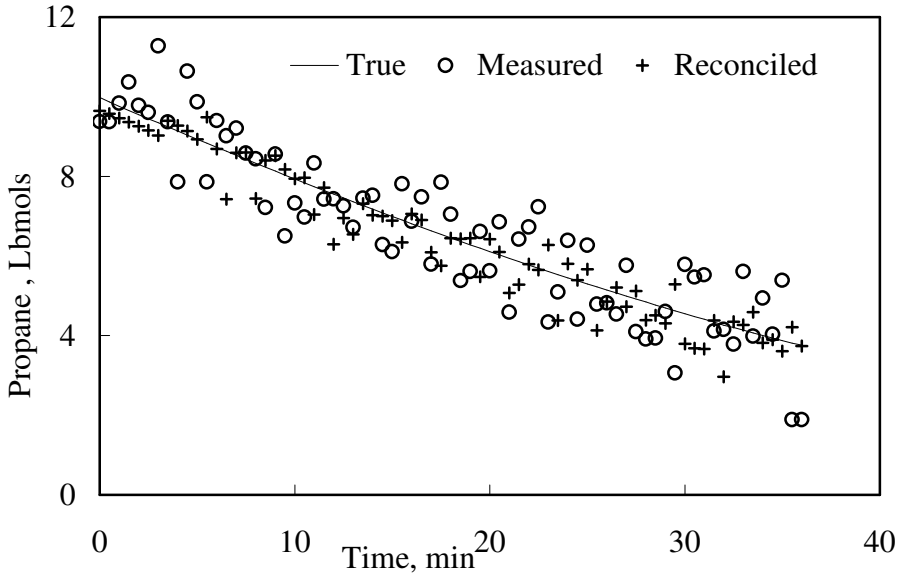


Figure 5-34. Reconciliation of Propane Content in the Distillate.

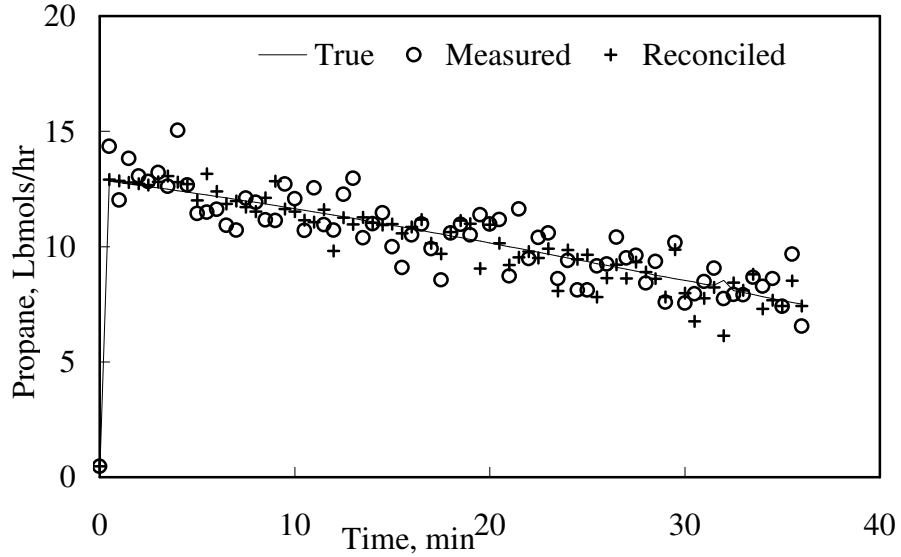


Figure 5-35. Reconciliation of Reboiler Heat Duty.

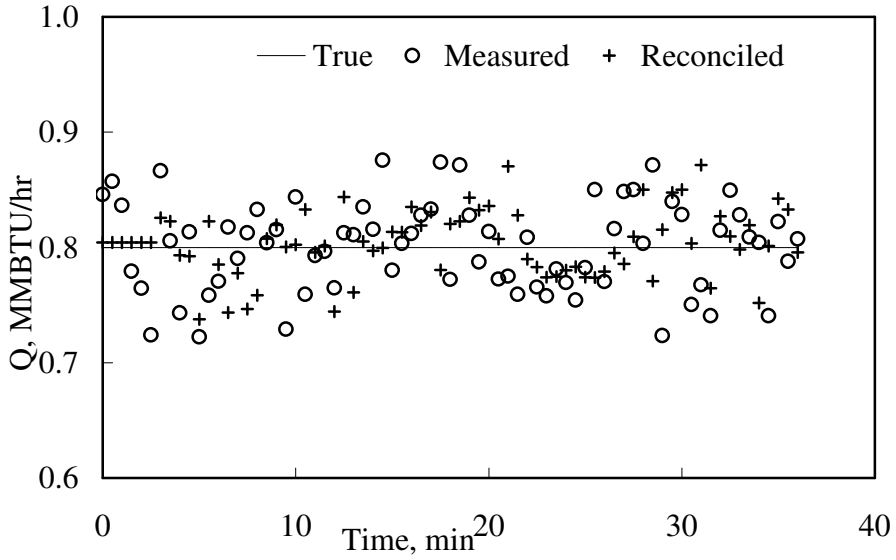


Figure 5-36. Reconciliation of Distillate Flow Rate.

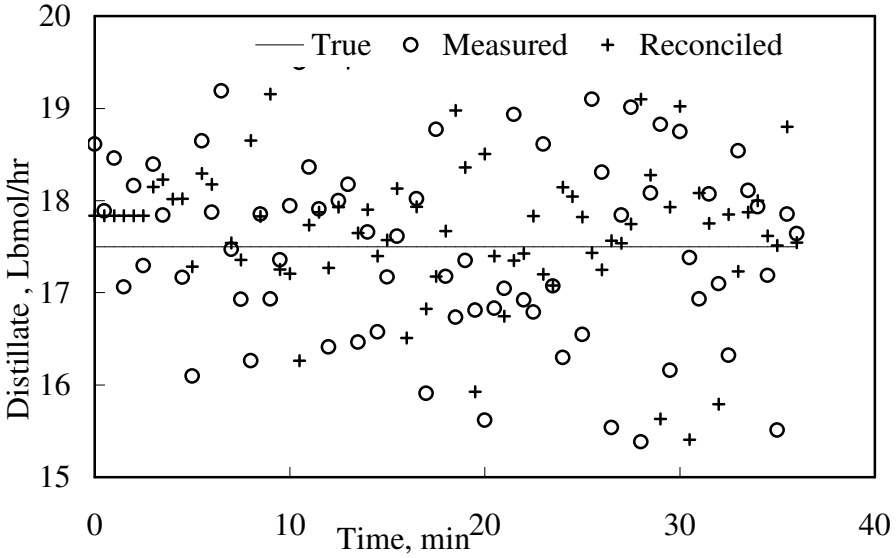


Figure 5-37. Reconciliation of Pentane Flow in the Distillate.

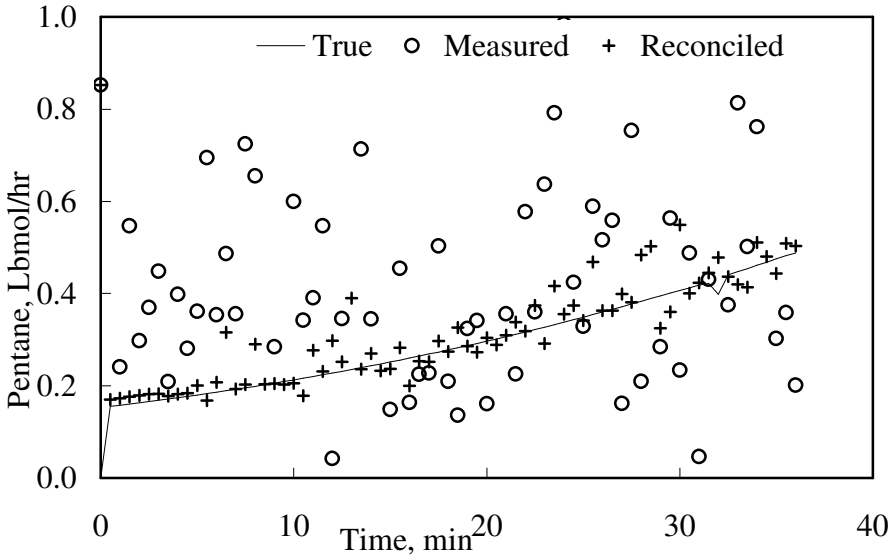


Figure 5-38. Reconciliation of Hexane Flow in the Distillate.

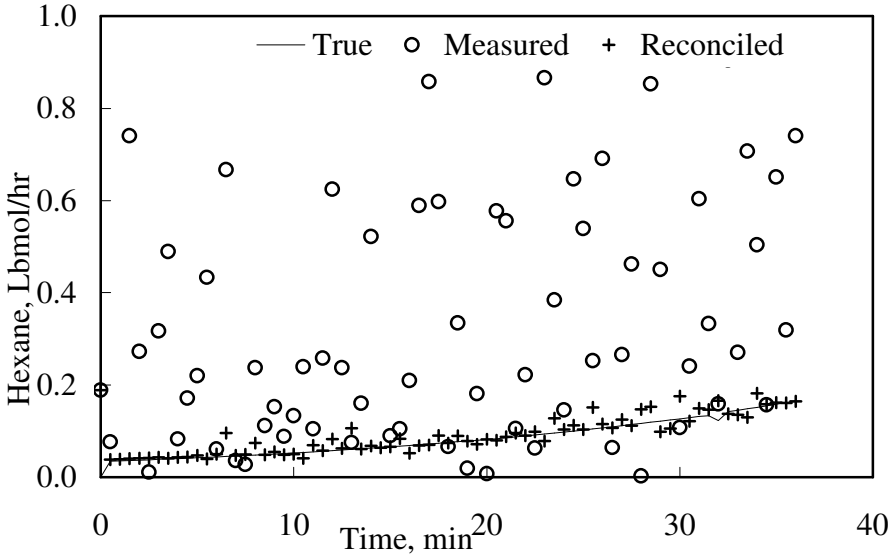


Figure 5-39. Butane in the Liquid Stream (Stream # 4) to the Reboiler.

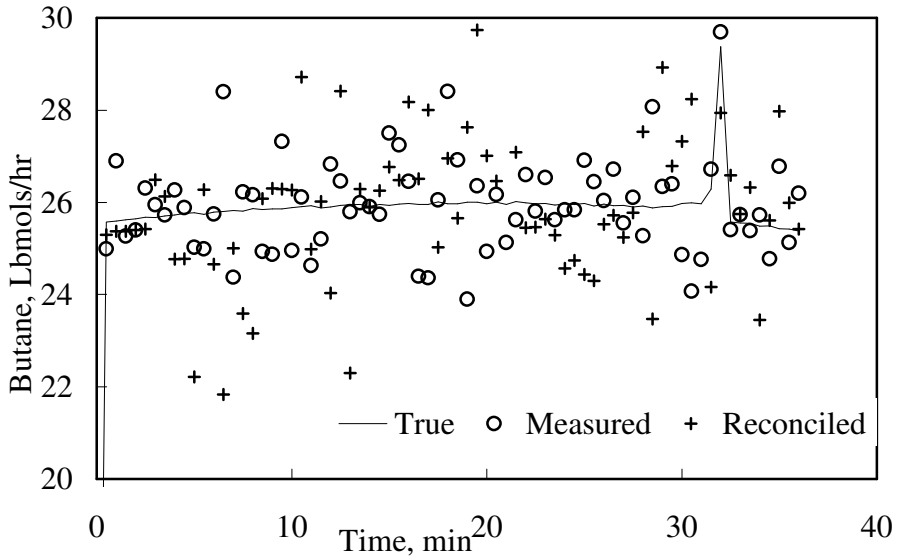


Figure 5-40. Propane in the Vapor Stream (Stream # 2) from the Reboiler to the Column.

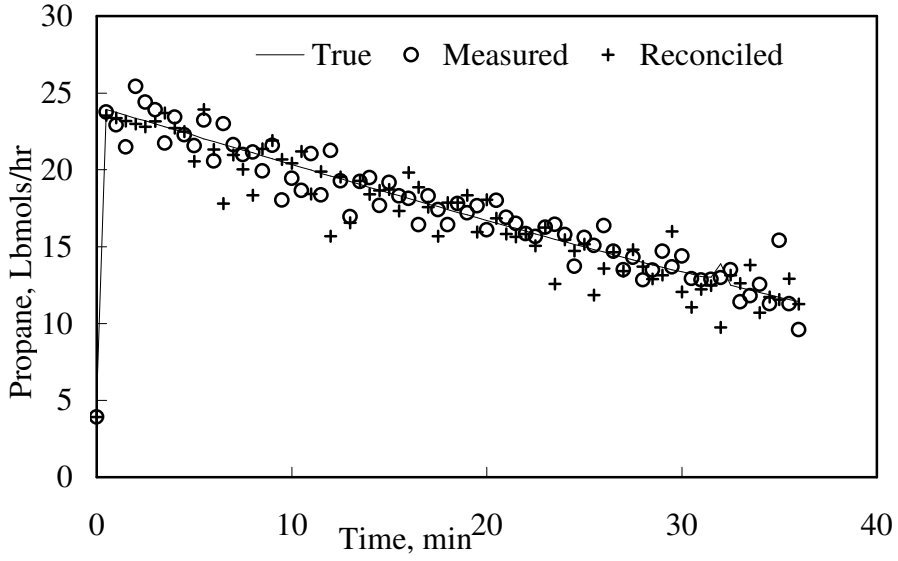
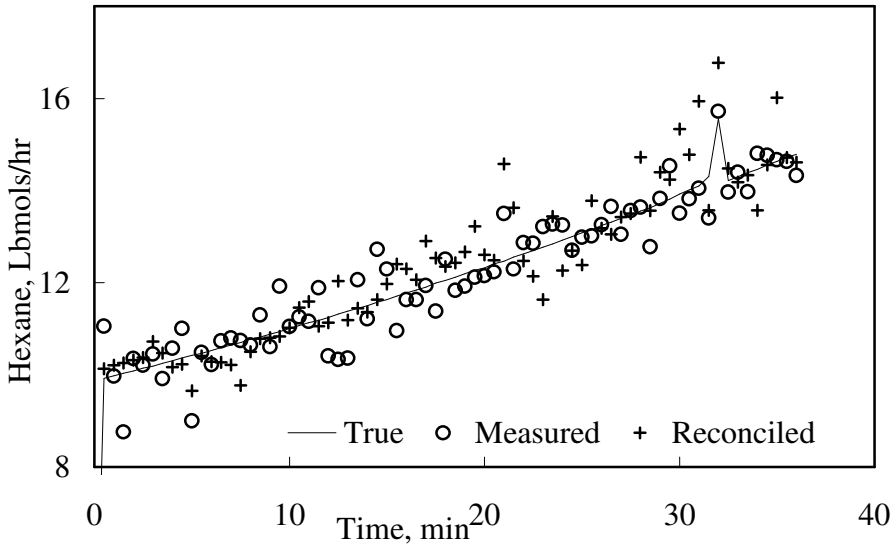


Figure 5-41. Hexane in the Vapor Stream to the Column



5.4.4 Results after Denoising

Denoising improves the reconciliation results for most of the measurements. After denoising the success rate and RMSE reduction increased by 5 % and 11 % on average. See Table 5-20, Figure 5-42 and Figure 5-43.

The average increase in RMSE reduction for the reboiler and distillate temperature and flow measurements is around 15 %. Pentane and hexane flow rates in the distillate stream are an exception. As explained in the previous section, the model was able to remove 70 to 90% error in the pentane and hexane flow rates without the aid of denoising. Denoising couldn't improve them any further. Figure 5-44 to 5-46 show examples of reconciliation trends from the remaining reboiler and distillate measurements. The increase in RMSE reductions for reboiler temperature, distillate rate and the heat duty are 15%, 24 % and 8% respectively. This improvement in reboiler temperature and distillate rate due to denoising can be seen in the Figure 5-44 and Figure 5-45.The improvement in the heat duty is hard to notice in the Figure 5-46 as the increased error reduction after denoising is only 8%.

The success rate and reduction in RMSE did not increase for the pressure measurements after denoising. Figure 5-47 compares the reconciled reboiler pressure in the presence and absence of denoising. The majority of the reconciled pressures do not change due to denoising. This is because the standard deviation in the measurements themselves was low (σ = 0.2 Psia in 102 Psia). This explains the lack of change in the success rate of reconciled pressures due to denoising. There were just three data points (highlighted in

Figure 5-47) where the reconciled pressures using denoising strayed away from the true behavior. These contributed to the increase in the RMSE of the reconciled reboiler pressures after using denoising.

The temperature and flow rate measurements for the intermediate streams see an average increase of 7% and 16% in the success rate and the RMSE reduction after denoising. Detailed trends of sample intermediate stream flow rates are given in Figure 5-48 to Figure 5-50. The propane (stream 2), butane (stream 4) and hexane (stream2) flows seen an average 20% increase in RMSE reduction. This is reflected in the figures.

The RBR methodology successfully reduced the error in the key measurements (i.e. conditions of reboiler & distillate streams) from a batch distillation operation. The reconciled values are better than the raw data for majority of the measurements. Denoising increased the error reduction and success rate in the reconciliation results for most of the measurements.

Figure 5-42. Percentage of Runs with Error Reduction, Comparison with and without Denoising

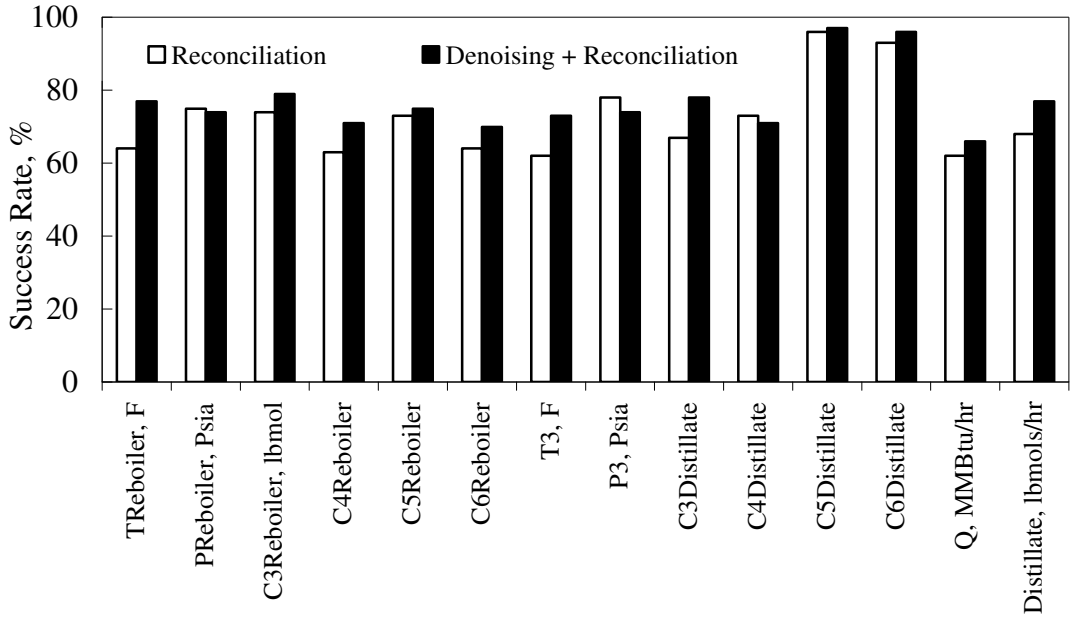


Figure 5-43. Reduction in RMSE, Comparison with and without Denoising

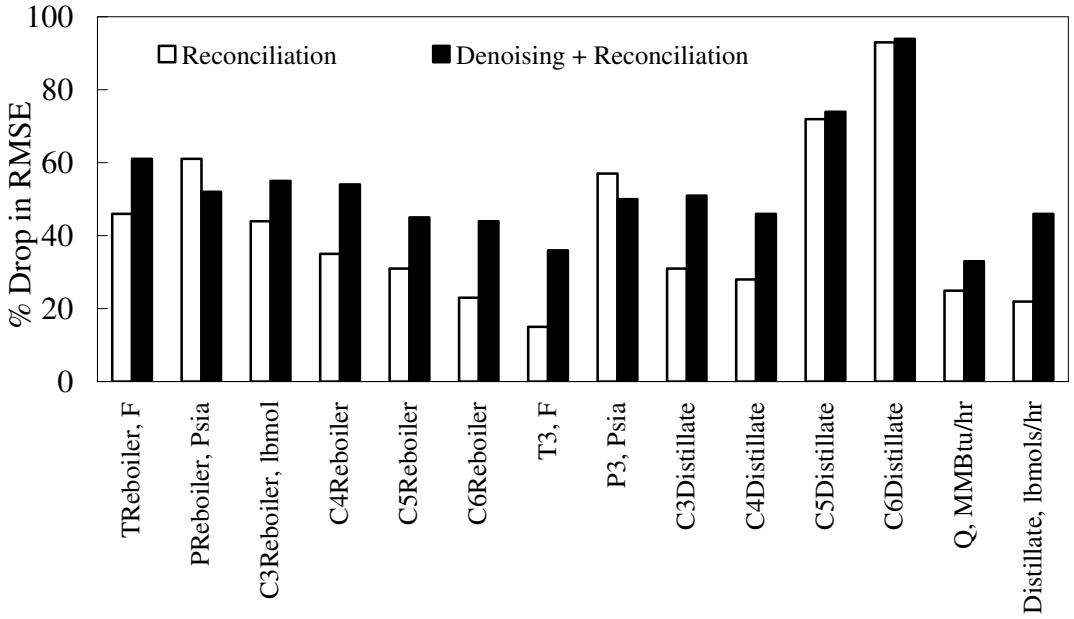


Table 5-20. Reconciliation of Batch Distillation Measurements with and without Denoising.

	% Success			RMSE Redcn		
	Raw Recn	Denois-ing	Den + Recn	Raw Recn	Denois-ing	Den + Recn
$T_{Reboiler}$, F	64	55	77	46	25	61
$P_{Reboiler}$, Psia	75	49	74	61	19	52
$C3_{Reboiler}$, lbmol	74	66	79	44	24	55
$C4_{Reboiler}$	63	56	71	35	24	54
$C5_{Reboiler}$	73	58	75	31	18	45
$C6_{Reboiler}$	64	55	70	23	23	44
T_2, F	73	53	78	46	16	56
P_2, Psia	77	53	77	61	24	52
$C3_2$	49	58	56	-27	23	-1
$C4_2$	42	52	53	-46	13	-26
$nC5_2$	66	56	74	32	19	40
$C6_2$	49	47	52	-24	9	-5
T_3, F	62	49	73	15	16	36
P_3, Psia	78	60	74	57	23	50
$C3_3$	67	49	78	31	17	51
$C4_3$	73	59	71	28	15	46
$C5_3$	96	56	97	72	21	74
$C6_3$	93	38	96	93	22	94
T_4, F	64	62	77	20	25	39
P_4, Psia	78	55	77	64	18	55
$C3_4$	60	56	64	2	22	17
$C4_4$	34	45	42	-79	17	-56
$C5_4$	68	52	75	43	17	49
$C6_4$	49	49	58	-35	15	-16
Q, MMBtu/hr	62	58	66	25	17	33
Distillate rate, lbmols/hr	68	56	77	22	25	46

Figure 5-44. Reconciliation of Reconciled Reboiler Temperature after Denoising.

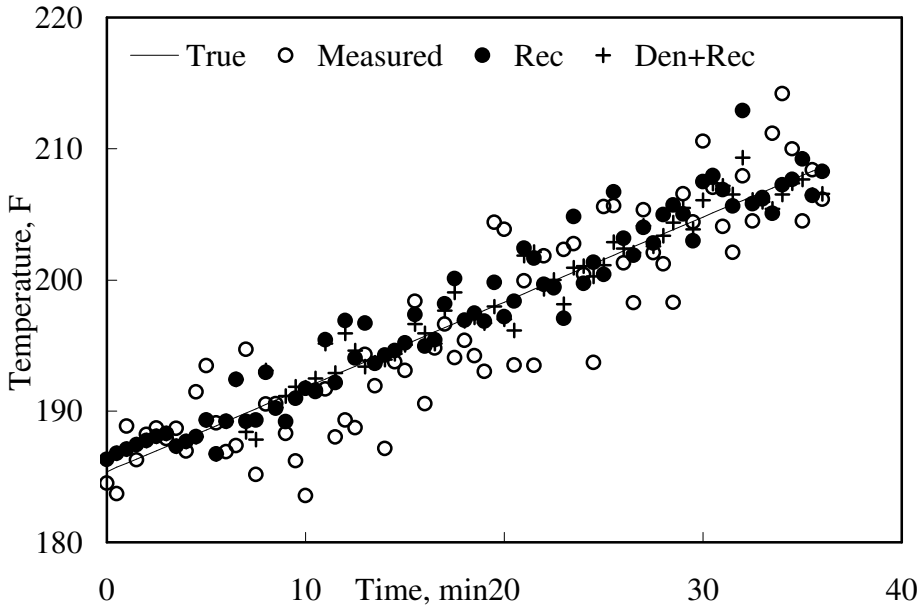


Figure 5-45. Comparison Reconciled Distillate Flow rate data with and without Denoising.

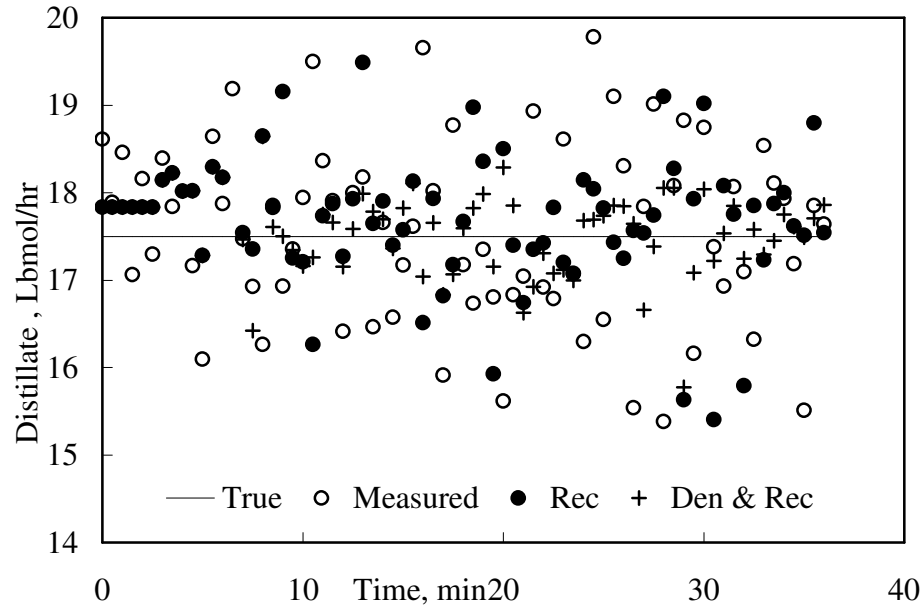


Figure 5-46. Comparison Reconciled Reboiler Heat Duty with and without Denoising

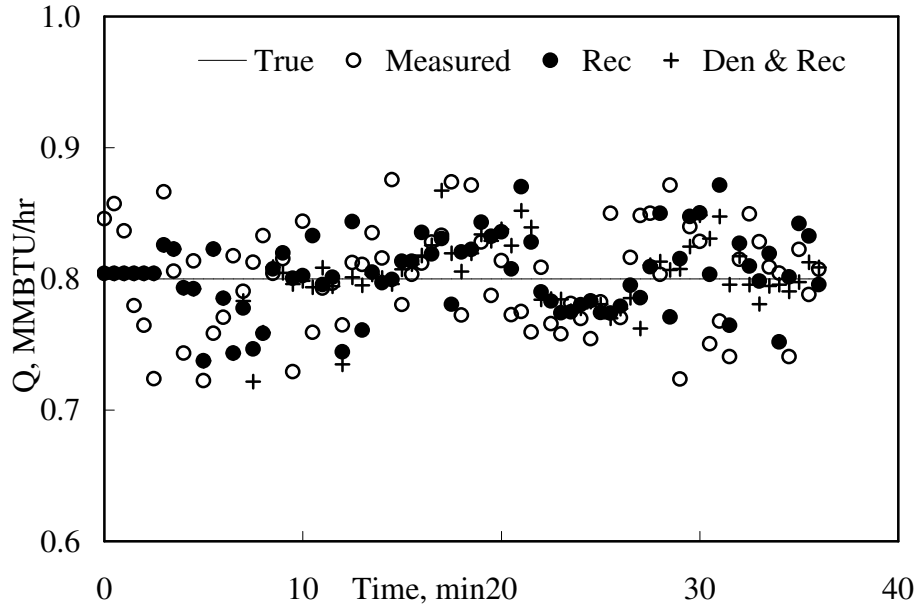


Figure 5-47. Comparison of Reconciled Reboiler Pressure With and Without Denoising.

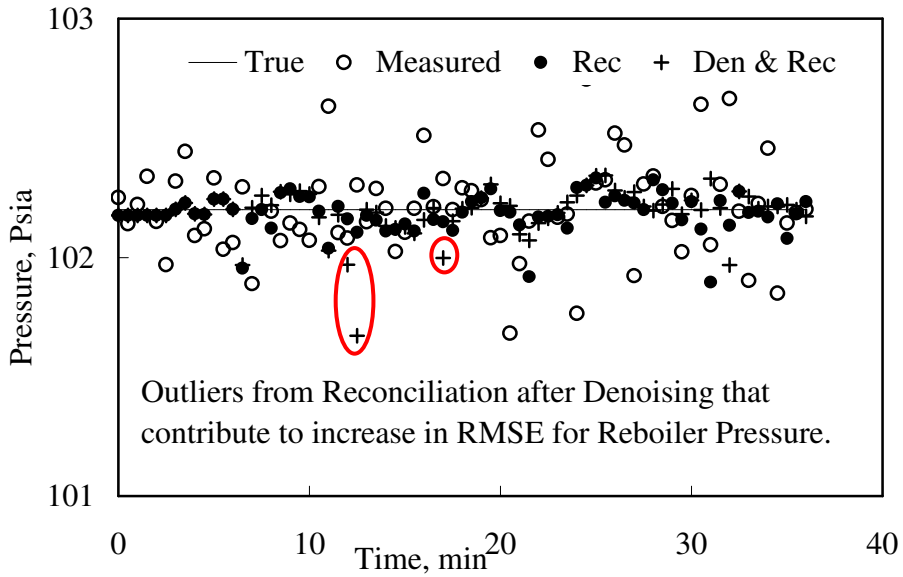


Figure 5-48. Comparison of Reconciled Propane flow rate in Reboiler Vapor Stream, with and without Denoising.

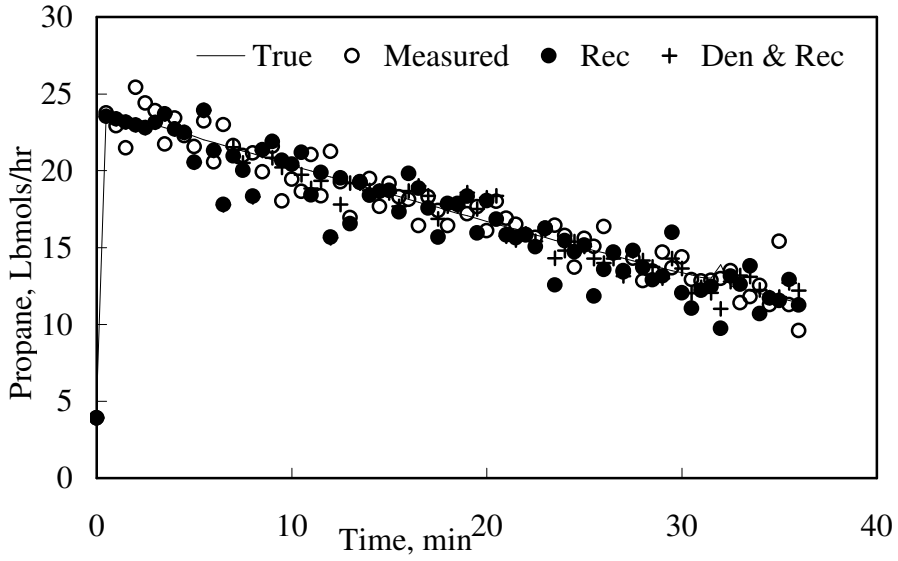


Figure 5-49. Comparison of Reconciled Butane flow rate in Liquid Stream to the Reboiler (Stream # 4), with and without Denoising.

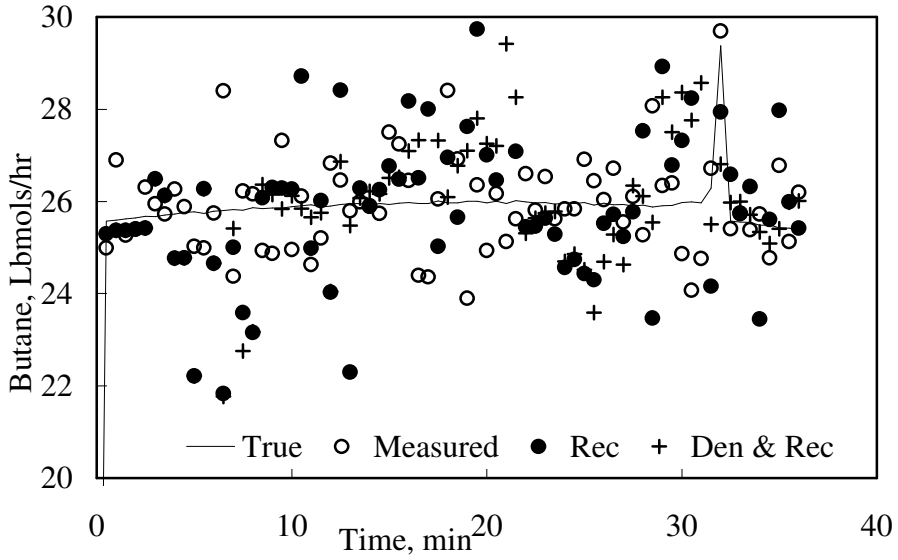
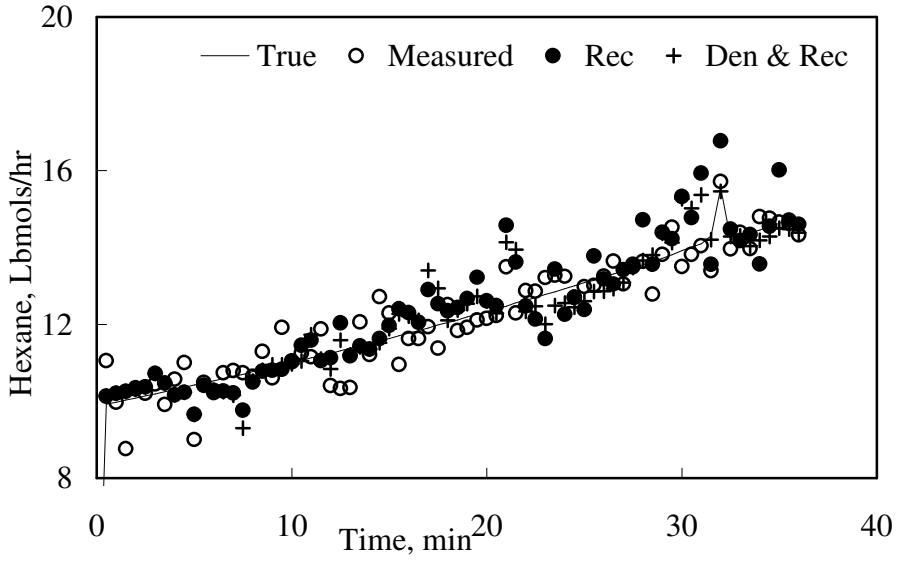


Figure 5-50. Reconciled Hexane flow rate in the Reboiler Vapor Stream, with and without Denoising.



6. Conclusions and Recommendations

This work presents a method to reconcile unsteady state plant measurements using process simulation software and optimization software. A substantial amount of programming is eliminated by using separate stand alone software for building process models and optimization purposes. The method is successful in reducing the normally distributed measurement error and in estimating any unknown process conditions. Reconciliation results are improved by applying wavelet denoising to the measurements.

The use of commercial simulation and optimization software allows the plant engineer to develop reconciliation programs faster. The strategy involves separating the model calculations from the optimization calculations. Implementing this approach requires communication between the process simulator and optimization software which can be achieved through OLE technology from Microsoft. Through the use of simulation software, complex thermodynamics and property methods, which critical to accurate modeling, can be easily built into the process model. The method can also be extended to process trains with multiple units, recycle streams and process controls. Separation of model and optimization calculations allows the optimization engine to be changed without affecting the model.

Wavelet denoising is used to reduce some of the random error present in the measurements. Its effectiveness is influenced by the process dynamics and the ratio of measurement value (i.e. signal) to the error (i.e. noise). There is a need for a method that can adaptively select the right wavelet order based on these conditions.

Three case studies are presented where measurements from a simple CSTR, a CPD dimerization reactor and a batch distillation column are reconciled using the proposed methodology. The measurements are computer generated and contain only random error. Results the proposed work are equivalent in accuracy to those from literature. Results from CPD Dimerization reactor and Batch Distillation show that the method can improve measurement accuracy for more complex processes. The proposed reconciliation was also found to estimate unmeasured process variables with reasonable accuracy.

The computation time for the proposed method is significantly higher than those from literature. The effect of increasing model complexity on the computation time must be explored and ways to reduce it must be studied. A framework for determining the maximum number of unmeasured variables which can be estimated by the reconciliation method must be developed.

The following are the recommendations from this work:

1. The proposed RBR (Rating Based Reconciliation) method must be extended to detect gross errors in the measurements.
2. RBR method must be tested for situations with multiple units, recycle streams and process controllers.
3. The effect of the optimization algorithm on the reconciliation results must be studied.
4. An adaptive wavelet denoising algorithm should be developed.

7. References

1. Addison. P.S., 2002. The Illustrated Wavelet Transform Handbook. Institute of Physics Publishing.
2. Albuquerque. J.S. and Biegler L.T., 1995. Decomposition Algorithms for Online Estimation with Nonlinear Models. Computers and Chemical Engineering, 19(10): 1031-1039.
3. Albuquerque. J.S. and Biegler L.T., 1995. Data Reconciliation and Gross-Error Detection for Dynamic Systems. AICHE Journal, 42(10): 2841-2856.
4. Alici. S. and Edgar. T.F, 2002. Nonlinear Dynamic Data Reconciliation via Process Simulation Software and Model Identification Tools. Ind. Eng. Chem. Res, 41(16): 3984-3992.
5. Becerra, V. M., P. D. Roberts, and G. W. Griffiths, 1998. Dynamic data reconciliation for sequential modular simulators. Control '98, UKACC International Conference on Control. Swansea, 1230-1235.
6. Biegler. L.T., 1984. Solution of Dynamic Optimization Problems by Successive Quadratic Programming and Orthogonal Collocation. Computers and Chemical Engineering, 8: 243-248.
7. Britt. H. I. and Luecke. R. H., 1973. The Estimation of Parameters in Non-linear Implicit Models. Technometrics, 15(2).
8. Carnahan. B, Luther. H. A. and Wilkes. J. O. 1969. Applied Numerical Methods. John Wiley & Sons, 100-105.

9. Canaday. W, 2003. Exploring the limits: Infinite dilution phase behavior of n-pentane in water and iso-pentane in water at 318 k. Masters Thesis, University of Kansas.

10. Chemstations, 2010. CHEMCAD Version 6 User Guide. www.Chemstations.com

11. Chen. J and Romagnoli. J. A, 1998. A Strategy for Simultaneous Dynamic Data Reconciliation and Outlier Detection. Computers and Chemical Engineering, 22(4/5): 559-562.

12. Cheng. L. W., 2003. Experimental measurement and analysis of the phase equilibria in the infinite dilution region for the systems N-Pentane-Acetonitrile, 2-Methyl-2-Butene-Acetonitrile, and Isopresne-Acetonitrile at 320K. Masters Thesis, University of Kansas.

13. Coleman, T.F. and Li. Y., 1994. On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear Minimization Subject to Bounds. Mathematical Programming, 67(2): 189-224.

14. Coleman, T.F. and Li. Y., 1996. An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds. SIAM Journal on Optimization, 6: 418-445.

15. Cuthrell, J. E. and Biegler, L. T., 1987. On the Optimization of Differential-Algebraic Process Systems. AICHE Journal, 33(8): 1257-1270.

16. Donoho, D.L., 1993. "Wavelet shrinkage and WVD: a ten minute tour," in Progress in Wavelet Analysis and Applications. Y. Meyer, S. Roques, Frontières Ed. pp. 109-128.

17. Donoho. D.L. and Johnstone. I.M., 1994. Ideal Spatial Adaptation by Wavelet Shrinkage. Biometrika, 81(3): 425 - 55.

18. Donoho. D.L. and Johnstone. I.M., 1995. Adapting to Unknown Smoothness via Wavelet Shrinkage. Journal of the American Statistical Association, 90 (432): 1200-1224.

19. Fugal, D.L., 2009. Conceptual Wavelets in Digital Signal Processing. Space and Signals Technical Publishing.

20. Graps. Amara., 1995. An Introduction to Wavelets. IEEE Comput. Sci. Eng., 2(2).

21. Haseltine, E. L., and Rawlings, J. B., 2005. Critical Evaluation of Extended Kalman Filtering and Moving Horizon Estimation. Ind. Eng. Chem. Res. 44(8): 2451-2460.

22. Hinkle, P., 2005. The Role of Wavelet Denoising in Improving Reconciliation and Interpretation in Plant Performance Analysis. Master's Dissertation. University of Kansas.

23. Holland. Charles. D. and Liapis., Athanasios. I., 1983. Computer Methods for Solving Dynamic Separation Problems. McGraw-Hill Book Company, 340-348.

24. Howat. J. N. and Howat. C. S., 1996. Infinite dilution activity coefficients and miscibility limit using a modified static total pressure method. Journal of Chemical Engineering Data. 41: 977-986.

25. Howat. C.S., 2004. Excursion Potential in Cyclopentadiene Reactors in the Purification of Isoprene. Paper 403af, AICHE Annual Meeting, San Francisco, CA.

26. Hubbard. B.B., 1998. The World According to Wavelets. A. K. Peters Ltd.

27. Jang, S.-S, Joseph. B. and Mukai. H., 1986. Comparison of Two Approaches to On-line Parameter and State Estimation of Nonlinear Systems. Ind. Eng. Chem. Process Des. Dev, 25: 809-814.

28. Jang, S.-S, Joseph. B. and Mukai. H., 1986. Comparison of Two Approaches to On-line Parameter and State Estimation of Nonlinear Systems. Ind. Eng. Chem. Process Des. Dev, 25: 809-814.

29. Kalman, R.E., 1960. A New Approach to Linear Filtering and Prediction Problem. Journal of Basic Engineering, 82(1): 35 - 45.

30. Karjala, T.W. and Himmelblau, D.M., 1994. Dynamic data rectification by recurrent neural networks versus traditional methods. AICHE Journal, 40:1865–1875.

31. Knepper. J. C. and Gorman. J.W., 1980. Statistical Analysis of Constrained Data Sets. AICHE Journal, 26(2): 262.

32. Kim, I.-W; Liebman, M. J. and Edgar, T. F., 1991. A Sequential Error-in-Variables Method for Nonlinear Dynamic Systems. Computers and Chemical Engineering, 15(9): 663-670.

33. Kopp, R.E., and Oxford, R. J., 1963. Linear Regression Applied to System Identification and Adaptive Control Systems. AIAA Journal. 1(10): 2300

34. Kuehn, D. R., and Davidson, H., 1961. Computer Control II: Mathematics of Control. Chemical Engineering Progress, 57(6): 44 - 47.

35. Lee, T., 2002. Measurement reconciliation and interpretation of a packed distillation column operation. Master's Thesis, University of Kansas, 2002

36. Liebman, M. J. and Edgar, T. F., 1988. Data Reconciliation for Nonlinear Processes. Preprint, AICHE Annual Meeting, Washington, DC.

37. Liebman, M. J., Edgar, T. F. and Lasdon, L. S., 1992. Efficient Data Reconciliation and Estimation for Dynamic Processes Using Nonlinear Programming Techniques. Computers and Chemical Engineering, 16(10/11): 963-986.

38. Luyben. W., 2002. Plantwide dynamic simulators in chemical processing and control, Volume 1. CRC press, 1st Edition.

39. Jansen. M., 2001. Noise Reduction by Wavelet Thresholding, Lecture Notes in Statistics. Springer.

40. MacDonald, R. J. and Howat, C.S., 1988. Data Reconciliation and Parameter Estimation in Plant Performance Analysis. AICHE Journal, 34(1): 1-8.

41. Mahidhara. D and Lasdon. L., 1989. An SQP Algorithm for Large Sparse Nonlinear Programs. Working Paper in the MSIS Department, The University of Texas at Austin.

42. Mathworks, 2010. MATLAB, Wavelet Toolbox Documentation.

43. Misiti, M., Y. Misiti, G. Oppenheim, J.M. Poggi., 1993. Analyse de Signaux Classiques Par Décomposition en Ondelettes. Revue de Statistique Appliquée, vol. XLI, no. 4, pp. 5–32.

44. Mohlenkamp. Martin and Pereyra. María Cristina, 2008. Wavelets, Their Friends and What They Can Do For You. European Mathematical Society Publishing House, Page 47.

45. Muske, K. R. and Edgar, T. F., 1997. Nonlinear State Estimation. Nonlinear Process Control, Edited by Henson, M. A. and Seborg, D. E. Prentice-Hall: 312.

46. Percival. D.B and Walden. A.T., 2006. Wavelet Methods for Time Series Analysis. Cambridge University Press.

47. Frontline Solvers, 2010. Premium Solver User Guide. Available at : http://www.solver.com/suppxlsguide.htm

48. Prata. D.M., Schwaab. M., Lima. E.L. and Pinto. J.C., 2009. Nonlinear Dynamic Data Reconciliation and Parameter Estimation through Particle Swarm Optimization: Application for an Industrial Polypropylene Reactor. Chemical Engineering Science, 64: 3953-3967

49. Ramamurthi, Y.; Sistu, P. B. and Bequette, B.W., 1993. Control Relevant Dynamic Data Reconciliation and Parameter Estimation. Computers and Chemical Engineering, 17(1): 41-59.

50. Robertson, D. G., Lee, J. H. and Rawlings, J. B., 1996. A Moving Horizon-Based Approach for Least-Squares Estimation. AICHE Journal, 42(8): 2209-2224.

51. Romagnoli. A. Jose. And Sanchez. M.A., 2000. Data Processing and Reconciliation for Chemical Process Operations. Academic Press.

52. Satuluri, M., 2003. Global Reconciliation of Infinite Dilution Activity Coefficient and Solubility Limit Phase Equilibria Measurements. Master's Dissertation, University of Kansas

53. Seinfeld, J. H., 1970. Optimal Stochastic Control of Nonlinear Systems. AICHE Journal, 6: 1016-1022.

54. Taswell, C., 2000. The What, How, and Why of Wavelet Shrinkage Denoising. IEEE Computing in Science and Engineering, 2(3): 12-19.

55. Vachhani, P.; Rengaswamy, R.; Gangwal, V. and Narasimhan, S., 2005. Recursive Estimation in Constrained Nonlinear Dynamical Systems. AICHE Journal, 51(3): 946-959.

56. Venkateswarlu, C., and Avantika, S., 2001. Optimal State Estimation of Multi Component Batch Distillation. Chemical Engineering Science, 56: 5771-5786.

57. Xie, W., and Rohani, S., 2001. Application of Extended Kalman Filter to a Batch Cooling Crystallizer. Separation Science and Technology, 36(13): 3049 - 3069.

58. Yildiz, U., Gurkan, U. A., Ozgen, C., and Leblebicioglu, K., 2005. State Estimator Design For Multi Component Batch Distillation Columns. Chemical Engineering Research and Design, 83(A5): 433 - 444.

8. Appendix I – Orthogonal Collocation

8.1 Method of Weighted Residuals

The method of Weighted Residuals (MWR, Holland and Liapis, 1983) is a numerical method used to solve differential equations. First, a trial solution is chosen such that the boundary conditions for the differential equation are satisfied. The trial solution generally is a polynomial function of the independent variable with unknown coefficients, *a*. In the case of dynamic process models, the independent variable is time, *t*. Next, the error that arises from substituting the trial solution into the differential equation is estimated. The error is called residual, *Rd(a, t)*. The residual is not a number. It is an algebraic equation made up of the unknown coefficients, *a*, and time, *t*.

Ideally, the coefficients, *a*, have to be such that the residual disappears at all points in the integration interval. However this is not possible. Instead the unknown parameters are chosen such that the net residual over the entire integration interval is zero. This is expressed mathematically in Equation (VIII-3).

$$\int_{b1}^{b2} Rd(a,t)dt \quad = \quad 0 \qquad \text{(VIII-3)}$$

Where [*b1*, *b2*] form the integration interval in which the trial solution is valid.

Gaussian quadrature using orthogonal polynomials (Carnahan, Luther and Wilkes, 1969) is used to evaluate Equation (VIII-3). In Gaussian quadrature the integral of any function is equal to the weighted sum of the function evaluations at certain unevenly spaced points

along the interval (Equation (VIII-4)). These points are called the collocation points. They are the roots of the associated orthogonal polynomial and can be easily calculated. The family of Gauss Jacobi orthogonal polynomials is frequently used in the literature.

$$\int_{b1}^{b2} f(t)dt \quad = \quad \sum_{i=0}^{n} w_i \cdot f(t_i) \qquad \text{(VIII-4)}$$

where, t_i are the collocation points, w_i are the weights and $(n+1)$ is the order of the associated orthogonal polynomial. The weights are calculated such that the integration is exact, if the order of the original polynomial, $f(t)$, is $2n+1$.

Substituting Equation (VIII-4) into (VIII-3), Equation (VIII-5) is obtained.

$$\int_{b1}^{b2} Rd(a,t)dt \quad = \quad \sum_{i=0}^{n} w_i \cdot Rd(a,t_i) \quad = \quad 0 \qquad \text{(VIII-5)}$$

The weights, w_i, are never equal to zero. Therefore Equation (VIII-5) gives rise to following equation.

$$Rd(a,t_i) \quad = \quad 0, \quad where\ i = 0\ to\ n \qquad \text{(VIII-6)}$$

Equation (VIII-6) results in a set of algebraic equations where the collocation points are known and the trial solution coefficients are unknown. The unknown coefficients are obtained by solving the residual equations. Notice that the number of unknown parameters in the trial solution must be equal to the order of the orthogonal polynomials (and therefore equal to the number of collocation points).

The Method of Weighted Residuals is also called orthogonal collocation because the parameters in the trial solution are calculated such that product of the residuals and the weights at the collocation points equals zero. In other words residuals are orthogonal to the weights in the way that x-axis and y-axis are orthogonal to one another in Cartesian system (Holland and Liapis, 1983).

8.2 Reconciliation Using Method of Weighted Residuals

MWR converts a differential equation into a set of algebraic equations. While applying this method to the Moving Window Estimation, each state variable is assigned one trial solution per sampling interval. This leads to n_c (i.e number of collocation points) residual equations per sampling interval for each state variable. These equations replace the differential equations in the process model.

Lagrange interpolating polynomials (Equation (VIII-7)) are frequently used in the literature (Liebman *et al*., 1992; Jang *et al*., 1986 and Vachhani *et al*., 2005) as the trial solution. This is because the constants in the polynomial function will have the same units as the dependent variable. In fact, they will represent the values of the dependent variables at time instances corresponding to the collocation points. This way the constants will have easily interpretable physical significance and therefore bounds can be placed on them during optimization.

$$\hat{x}(t) = \sum_{i=0}^{nc+1} w_i(t) \cdot \hat{x}(t_i) \qquad \text{(VIII-7)}$$

where $w_i(t) = \prod_{j=0, j\neq i}^{nc+1} \frac{t - t_j}{t_i - t_j}$, t_0 and t_{nc+1} are the start and end points in the interval and t_1 to t_{nc} are the collocation points in the interval $[t_0, t_{nc+1}]$. t_0 to t_{nc+1} are known while $\hat{x}(t_i)$ are unknown.

After choosing the interpolating polynomial, the order of the polynomial is determined. Higher order polynomials result in larger optimization problems. It is also not known whether increasing the order the polynomials leads to better accuracy in the results. The collocation points are the roots of the orthogonal polynomials converted from the interval [0, 1] to $[t_0, t_{nc+1}]$ by multiplying them with Δt.

Equation (VIII-8) represents a typical differential equation involving only state variables. Differentiating Equation (VIII-7) gives rise to the Equation (VIII-9). The right hand sides of equations (VIII-8) and (VIII-9) must be equal at all the collocation points. This is expressed in Equation (VIII-10). In this equation, the differential term $\frac{dw_i(t)}{dt}$ can be calculated and it is only the $\hat{x}(t_i)$ (where i= 1 to n_c) that are unknown. This way the differential equation is converted into multiple algebraic equations.

$$\frac{d\hat{x}(t)}{dt} = f(\hat{x}(t), t) \qquad \text{(VIII-8)}$$

$$\frac{d\hat{x}(t)}{dt} = \sum_{i=0}^{nc+1} \frac{dw_i(t)}{dt} \cdot \hat{x}(t_i) \qquad \text{(VIII-9)}$$

$$Rd(\hat{x}(t),t) = f(\hat{x}(t),t) - \sum_{i=0}^{nc+1} \frac{dw_i(t)}{dt} \cdot \hat{x}(t_i) = 0 \qquad \text{for all } t = t_1 \text{ to } t_{nc} \qquad \text{(VIII-10)}$$

After converting the differential equation, the resulting multiple algebraic equations are added as constraints to the optimization problem. Note that the algebraic equations in the process model also have to be satisfied at the collocation points. So, the number of algebraic equations will increase.

Assuming that the measurements vector, Z, consists of only the state variables, *x*, the reconciliation problem according to the simultaneous approach is defined by Equations (VIII-11) to (VIII-13).

$$\min_{\hat{x}_{i,k}} \sum_{i=1}^{n_m} \sum_{k=1}^{(n_c+1)\cdot h+1} \lambda_{i,k} \left(\frac{x_{i,k} - \hat{x}_{i,k}}{\sigma_{i,k}} \right)^2 \qquad \text{(VIII-11)}$$

Such that $Rd(\hat{x}(t)) = 0$ (VIII-12)

$$g(\hat{x}(t)) = 0 \qquad \text{(VIII-13)}$$

Where $\lambda_{i,k} = 0$ for all values of *i*, and k corresponding to the collocation points.

$\lambda_{i,k} = 1$ for all values of *i*, and k corresponding to time instances where actual measurements are available.

Note that in Equation (VIII-11), the unknown values of the state variables at the collocation points are added to the list of the optimization variables. However they are not

included in the objective function as there are measurements available at the collocation points.

9. Appendix II - Comparing Traditional Batch Distillation with Alternative Configuration

1. Use Chemcad Batch Distillation Example as the foundation.
2. Modify the example to ensure a positive distillate temperature.
3. Build an equivalent simulation using Dynamic Column (I will call this the alternative configuration from now on)
4. Arrive at total reflux conditions using the alternative configuration.
5. Validate the total reflux results.
6. Run Batch Distillation operation using the alternative configuration.
7. Compare the results from traditional Batch Distillation Simulation to those from the alternative configuration.

The results from steps 5 and 7 are documented in the following pages.

Alternative Configuration for Total Reflux Conditions

Figure 1. Chemcad Flow Sheet Used for Total Reflux Conditions

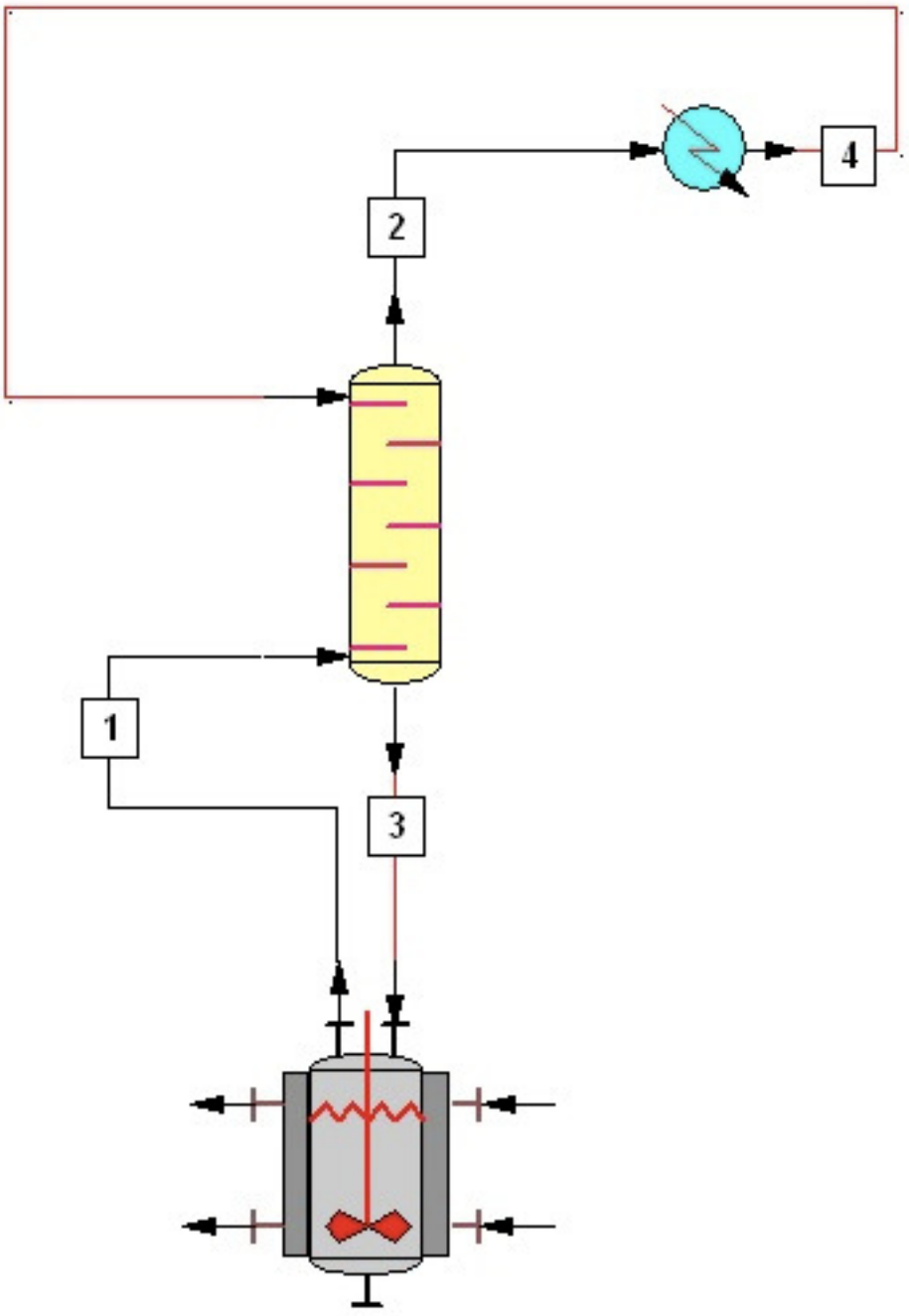


Table 1. Batch Reactor and Dy Column Specs

Batch reactor specifications:

Parameters	Values Choosen	Parameters	Values Choosen
Thermodynamics		*Reactor Specs*	
K-value Model	PR	Reactor Vol	250 ft^3
Enthalpy model	PR	Dia	4.8 ft
heat capacity	Polynomial	Wall thickness	0.0833 ft
		Wall density	490.75 lbs/ft^3
Initial Charge		Wall cp	0.1125 Btu/lb-F
Temp, F	184.63	Wall ther con	27.5 Btu/hr-ft-F
Press, Psia	102.2	Impeller Dia	1 ft
C3, lbmols	10	Impeller speed	0.00001 hz
C4	30		
C5	10	Vapor drawn to keep P constant	
C6	50		
General Info			
Reactor phase	Mixed		
Thermal mode	Heat duty		
Heat duty	0.15 MMBtu/hr		
Constant P	102.2 Psia		
Neglect Compression & Expansion			

Dynamic Column Specifications

Parameters	Values Choosen	Parameters	Values Choosen
General		*Startup Cond's*	
# of Stages	2	MaxAccholdup	0 ft^3
Feed stage for 4	1	Initial holdup	0 ft^3
Feed stage for 1	2		
Top pressure	102 Psia		
		Remaining options are left the way they were.	
Heat & MB spec			
Condenser duty	No Condenser		
No reboiler			
General Info			
Initial Column conditions	Dry tray startup		
Ignore liquid and vapor holdup			

Simple Heat Ex

Pressure Drop	2 Psia	Vapor Fraction of 4	0.00001
Holdup	1 ft^3		

Table 2. Conditions at total reflux with reboiler duty of 0.15 MMBtu/hr

Variable	Alternate configuration	BatchDistillation At (t = 0), total reflux, Δt = 0.1min	Mole Fractions	
Distillate			Alternate	BatchDistillation
Temperature, F	62.88	62.58		
Pressure, Psia	100	100		
C_3H_8, lbmols of reflux + Distillate	17.97	Unavailable*	0.8671868	0.87
C_4H_{10}	2.66		0.12	0.12
C_5H_{12}	0.07		0.003	0
C_6H_{14}	0.01		0.0006	0
Bottoms				
Temperature, F	187.15	186.4		
Pressure, Psia	102.2	102.2		
C_3H_8, lbmols	9.14	9.38	0.09	0.09
C_4H_{10}	29.60	29.91	0.30	0.3
C_5H_{12}	9.95	10	0.10	0.1
C_6H_{14}	49.90	50	0.50	0.5

Batch Distillation Operation Using the Alternative Configuration

Figure 2. Chemcad Flow Sheet Used for Batch Distillation Simulation

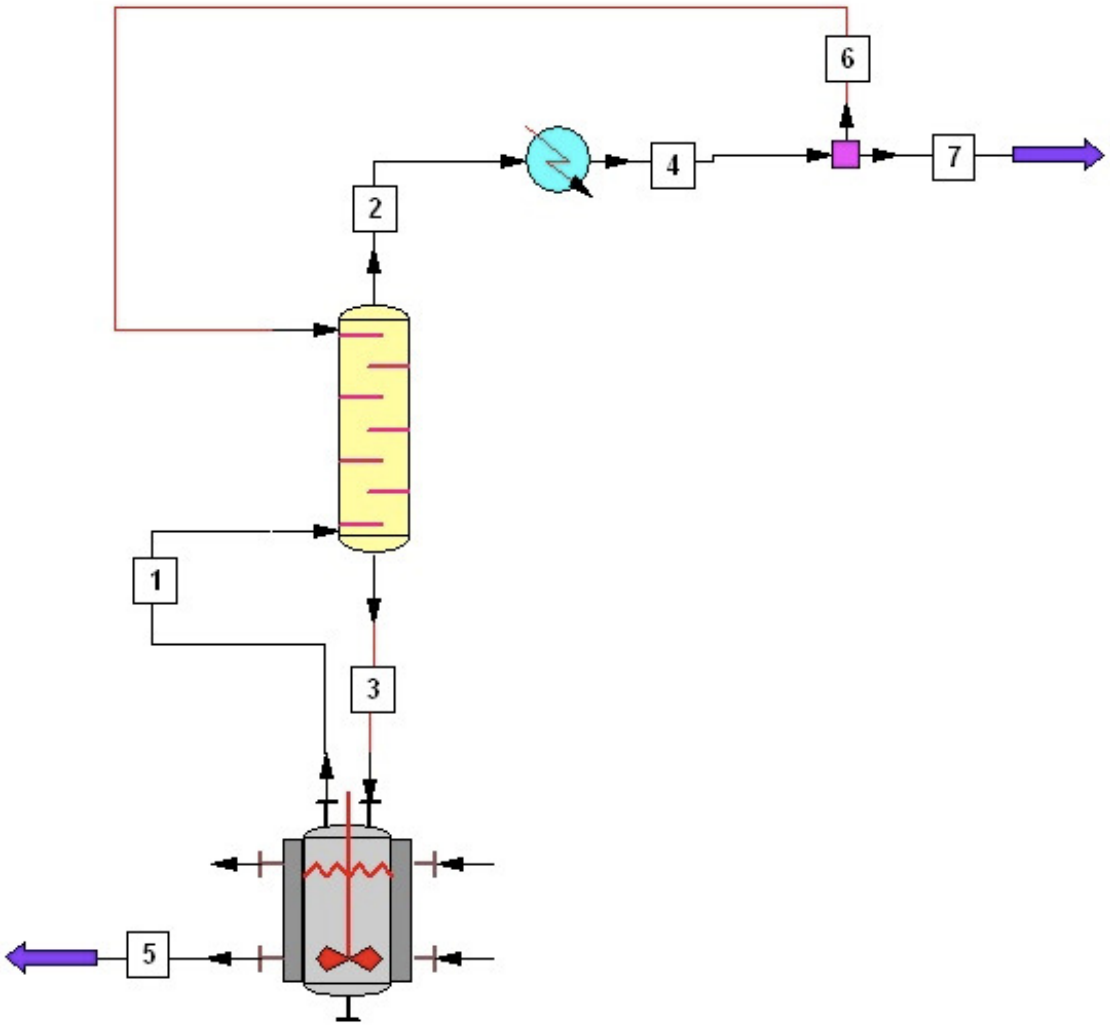


Table 3. Batch Reactor and Dy Column Specs

Batch Reactor

Parameters	Values Choosen	Parameters	Values Choosen
Thermodynamics		*Reactor Specs*	
K-value Model	PR	Reactor Vol	250 ft^3
Enthalpy model	PR	Dia	4.8 ft
heat capacity	Polynomial	Wall thickness	0.0833 ft
		Wall density	490.75 lbs/ft^3
Initial Charge		Wall cp	0.1125 Btu/lb-F
Temp, F	187.15	Wall ther con	27.5 Btu/hr-ft-F
Press, Psia	102.2	Impeller Dia	1 ft
C3, lbmols	9.36	Impeller speed	0.00001 hz
C4	29.90		
C5	9.99	Vapor drawn to keep P constant	
C6	49.99		
General Info			
Reactor phase	Mixed		
Thermal mode	Heat duty		
Heat duty	**0.8 MMBtu/hr**		
Constant P	102.2 Psia		
Neglect Compression & Expansion			

Dy Column

Parameters	Values Choosen	Parameters	Values Choosen
General		*Colmn holdups*	
# of Stages	2 (no condnser)	Condenser	1 ft^3
Feed stage for 6	1	Stage	0.01 ft^3
Feed stage for 1	2		
Top pressure	102 Psia	Remaining options are left the way they were.	
Colmn Δ P	0.2 Psia		
Heat & MB spec			
No Condenser			
No reboiler			
General Info			
Initial Column conditions		Dry tray startup	
Ignore vapor holdup & Constant liquid holdup			

Simple Heat Ex

Pressure Drop	2 Psia	Vapor Fraction of 4	0.00001
Holdup	1 ft^3		

Divider

Split Based on	Flow ratio	Output stream 6	0.8
		Output stream 7	0.2

Table 4. Initial conditions in streams 1 to 7

Stream	1	2	3	3	5	6	7
Temp, F	187.15	80.13	118.14	62.88	187.15	62.88	62.88
Pres Psia	102.2	102	102	100	102.2	100	100
V/F	1	1	0	0	0	0	0
Enthalpy	-0.77	-0.96	-0.86	-1.11	0	-0.89	-0.22
Total Flow	14.79 lbmol/h	20.64 lbmol/h	13.86 Lbmol/h	20.72 lbmol/h	0 lbmol/h	16.58 lbmol/h	4.14 Lbmol/h
	lbmol/h	lbmol/h	Lbmol/h	lbmol/h	lbmol/h	mole frac	mole frac
C3	4.97	17.72	4.35	17.97	0	0.86	0.86
C4	6.60	2.83	6.29	2.66	0	0.13	0.13
C5	1.03	0.081	1.01	0.073	0	0.003	0.003
C6	2.20	0.015	2.19	0.013	0	0.000	0.000

Note that the above initial conditions are from the total reflux conditions.

The differences between the batch distillation results and the results from dynamic column flow sheet are plotted in figures 3 to 16.

Fig 3.

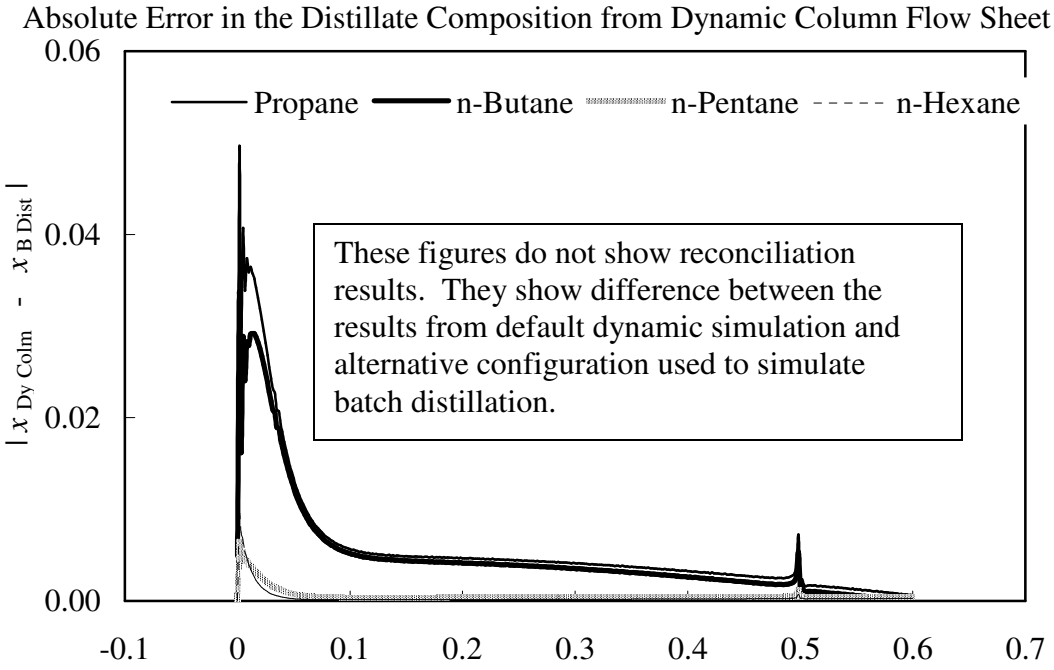

Absolute Error in the Distillate Composition from Dynamic Column Flow Sheet
Propane
n-Butane
n-Pentane
n-Hexane
These figures do not show reconciliation results. They show difference between the results from default dynamic simulation and alternative configuration used to simulate batch distillation.
| x Dy Colm - x B Dist |
0.06
0.04
0.02
0.00
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7

Fig 4.

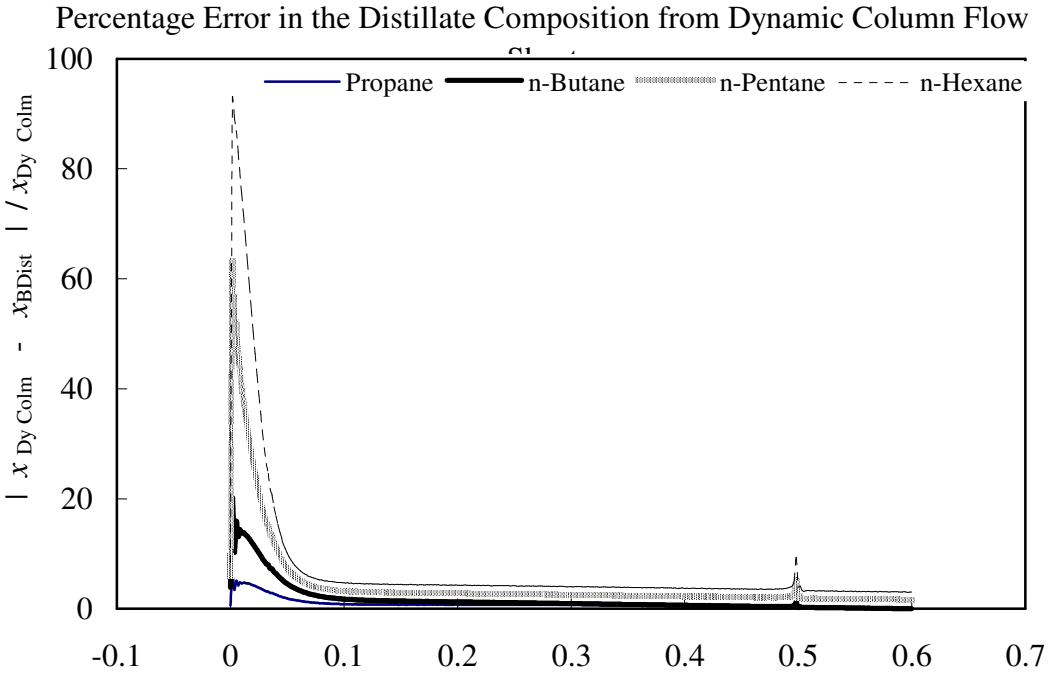

Percentage Error in the Distillate Composition from Dynamic Column Flow
Propane
n-Butane
n-Pentane
n-Hexane
| x Dy Colm - xBDist | / xDy Colm
100
80
60
40
20
0
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7

Figure 5.

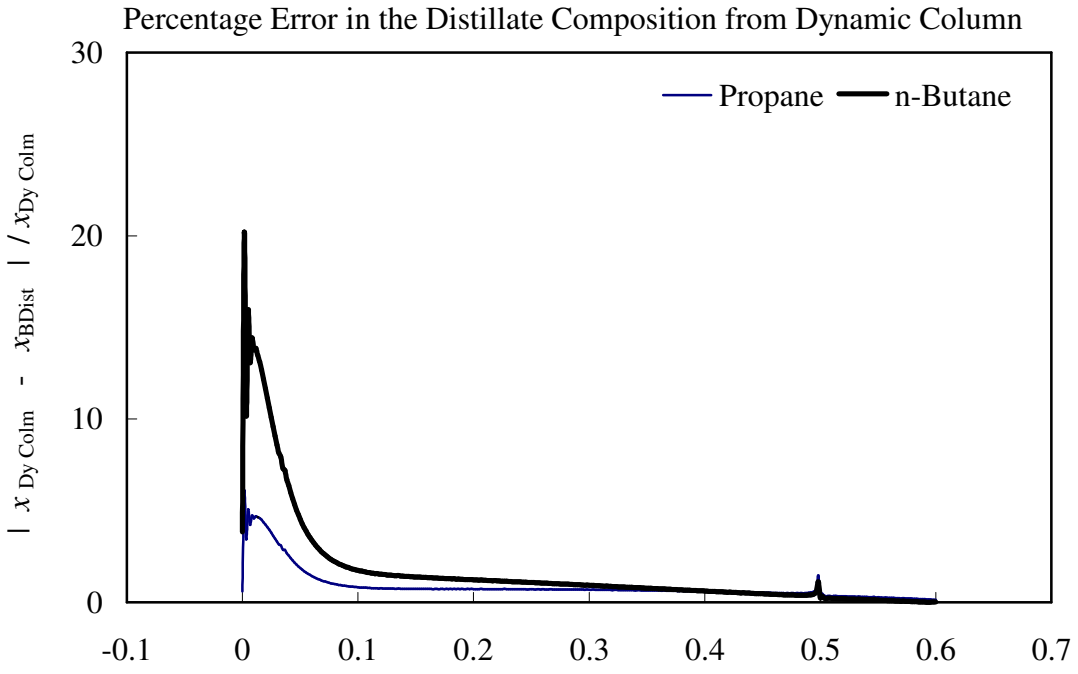


Figure 6.

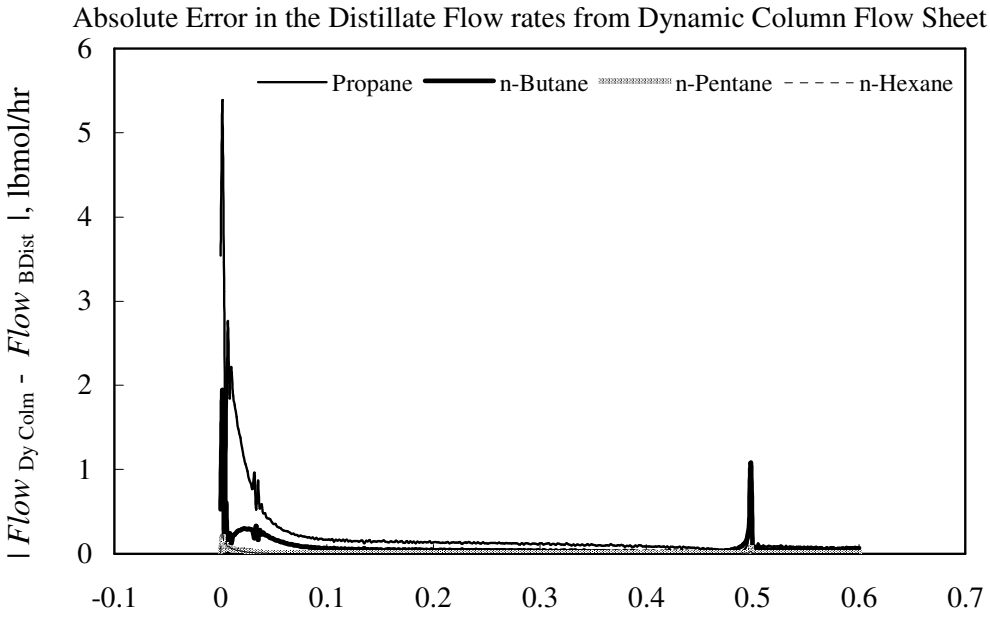


Figure 7.

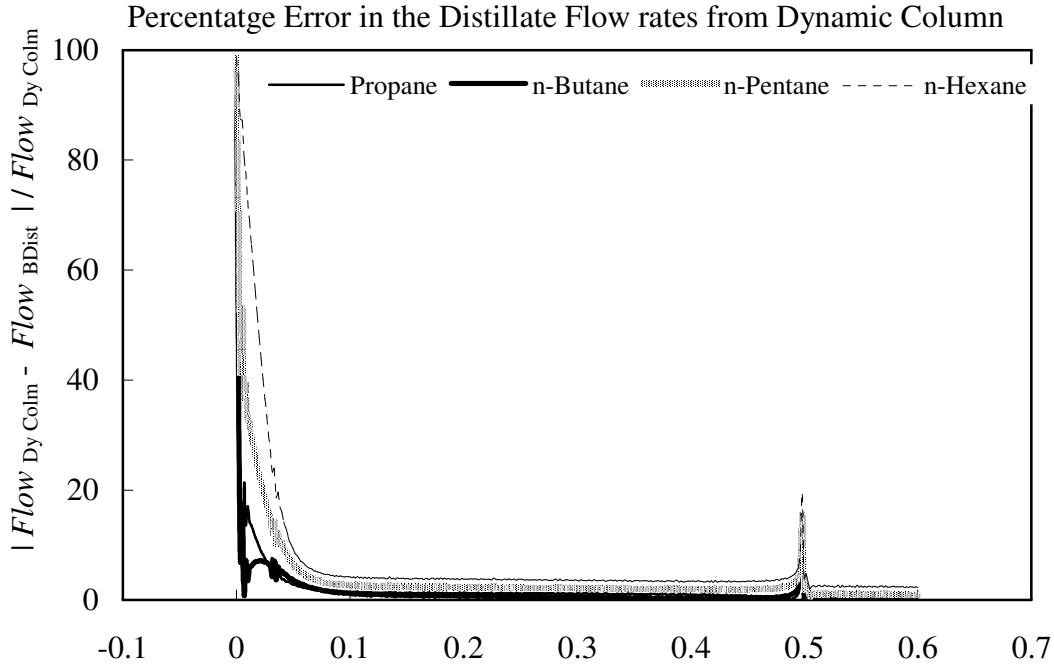


Figure 8.

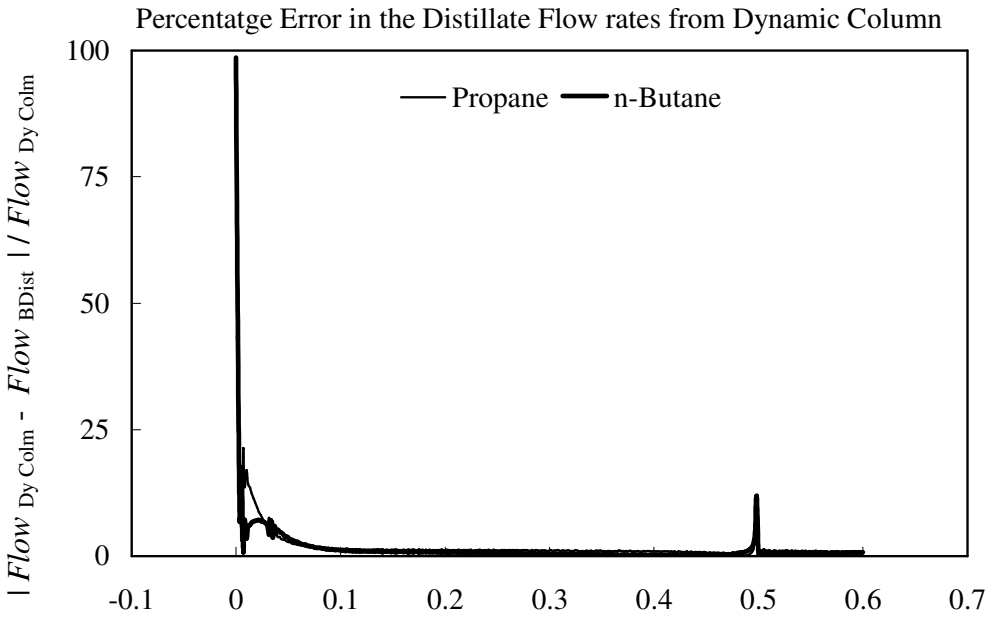


Figure 9.

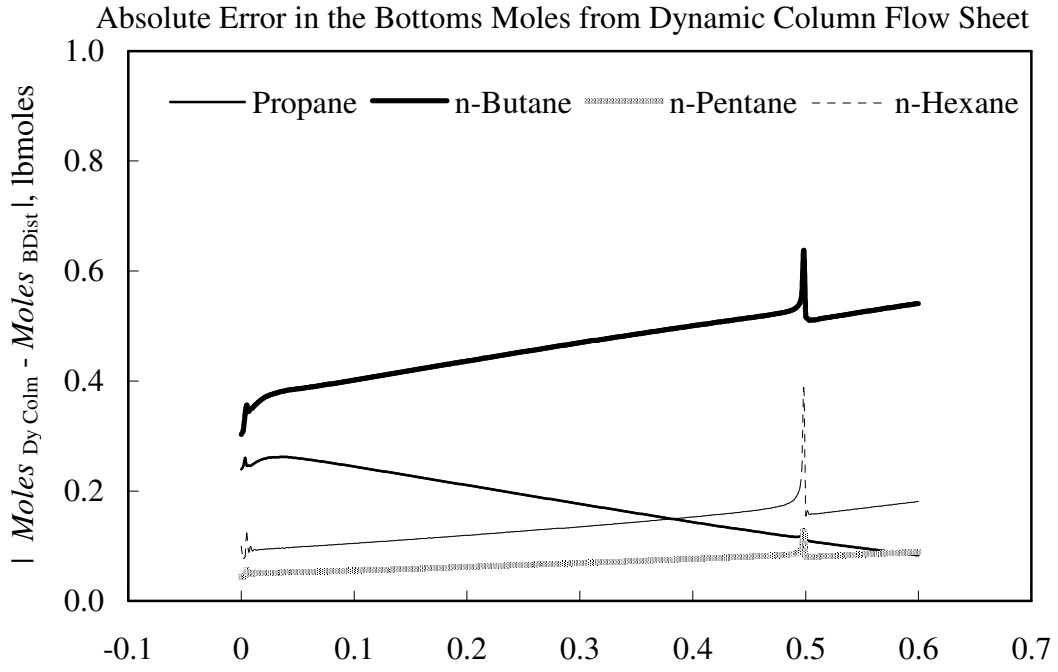


Figure 10.

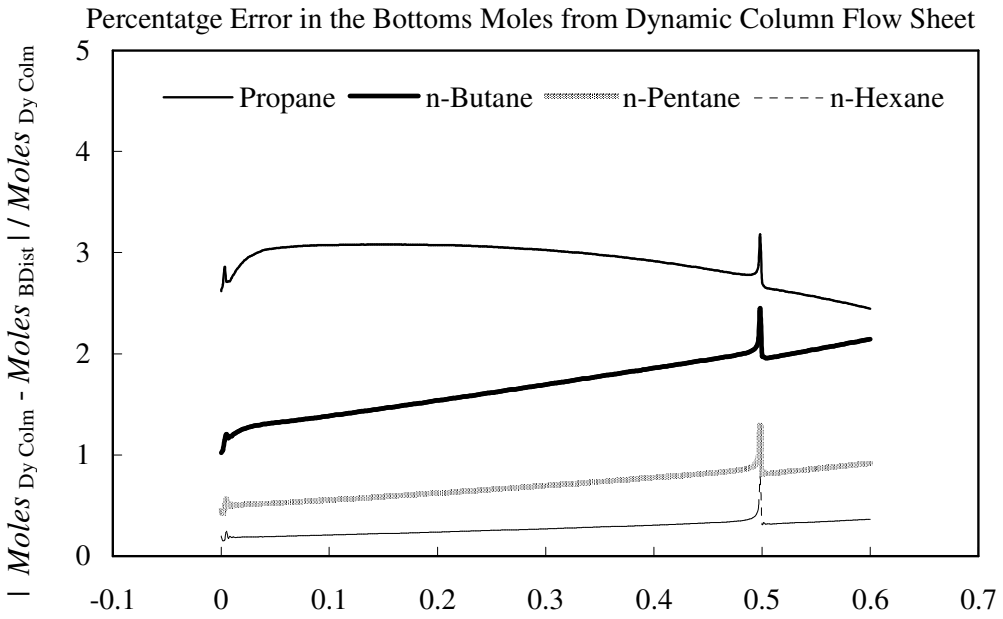


Figure 11.

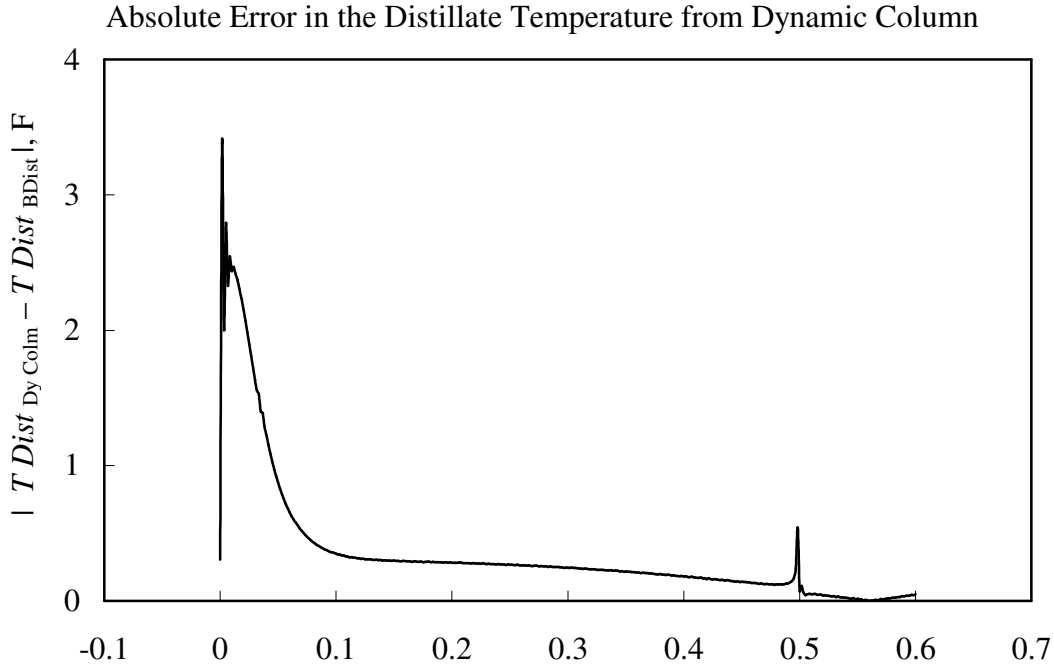


Figure 12.

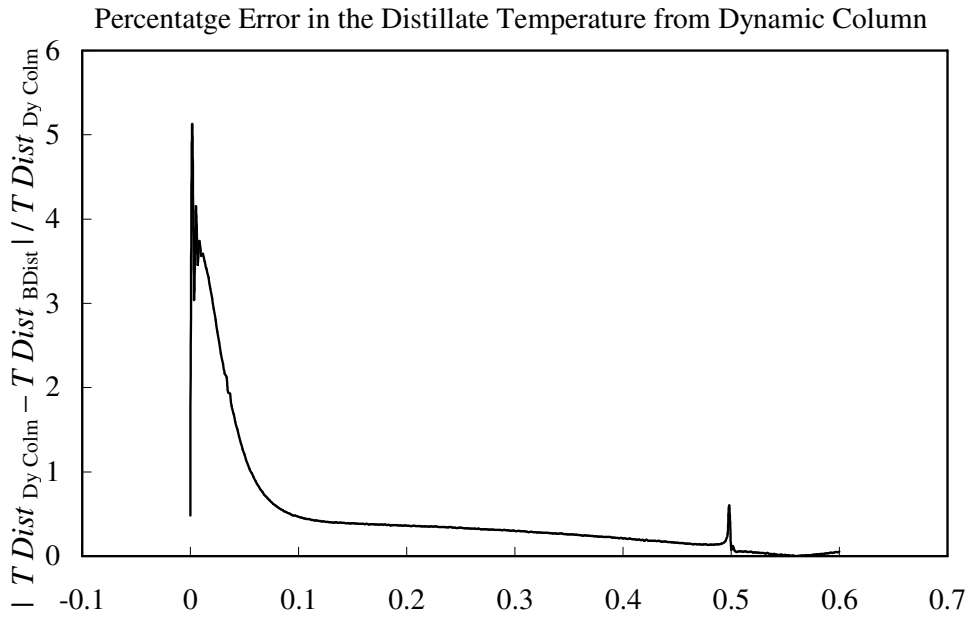


Figure 13.

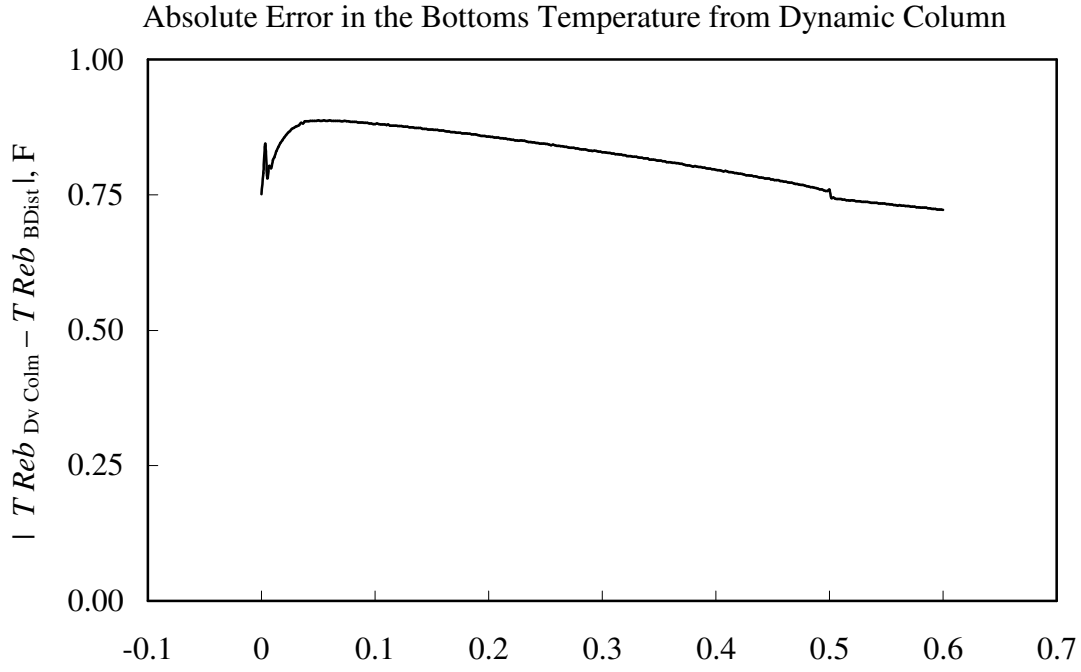


Figure 14.

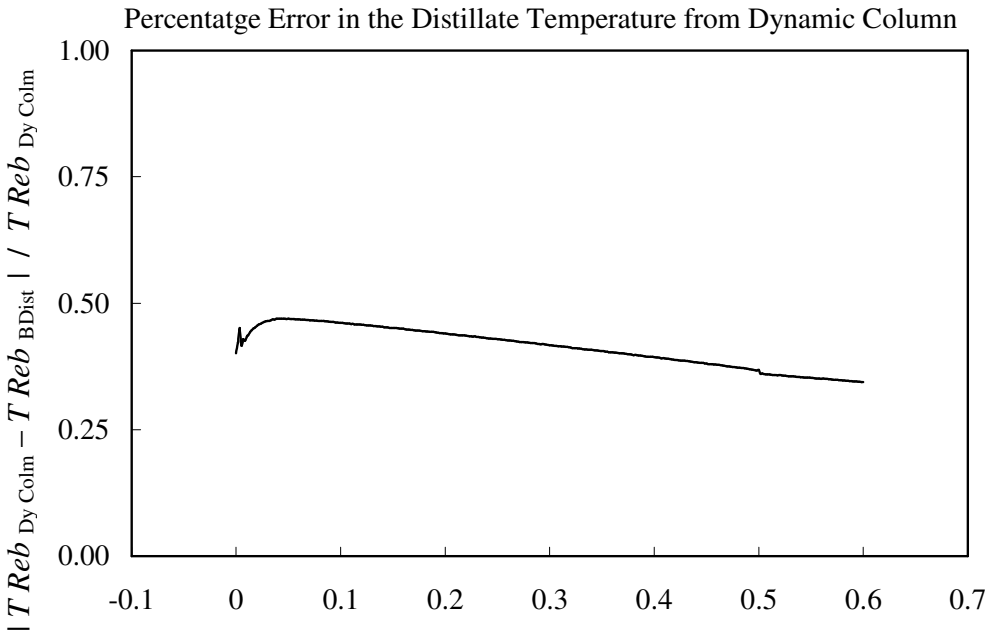


Figure 15.

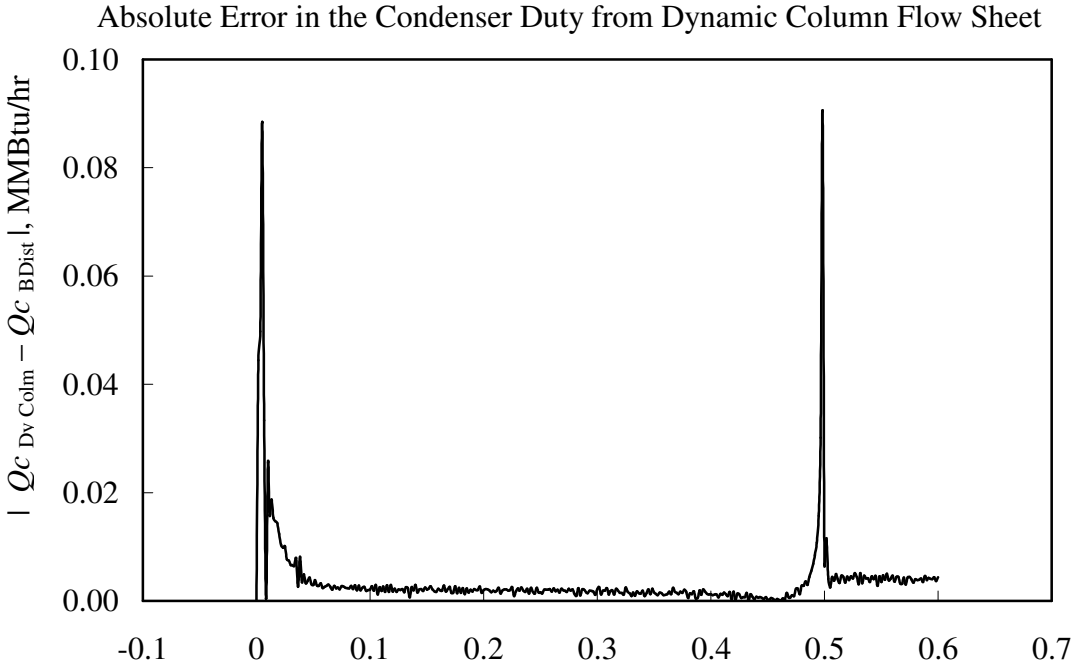
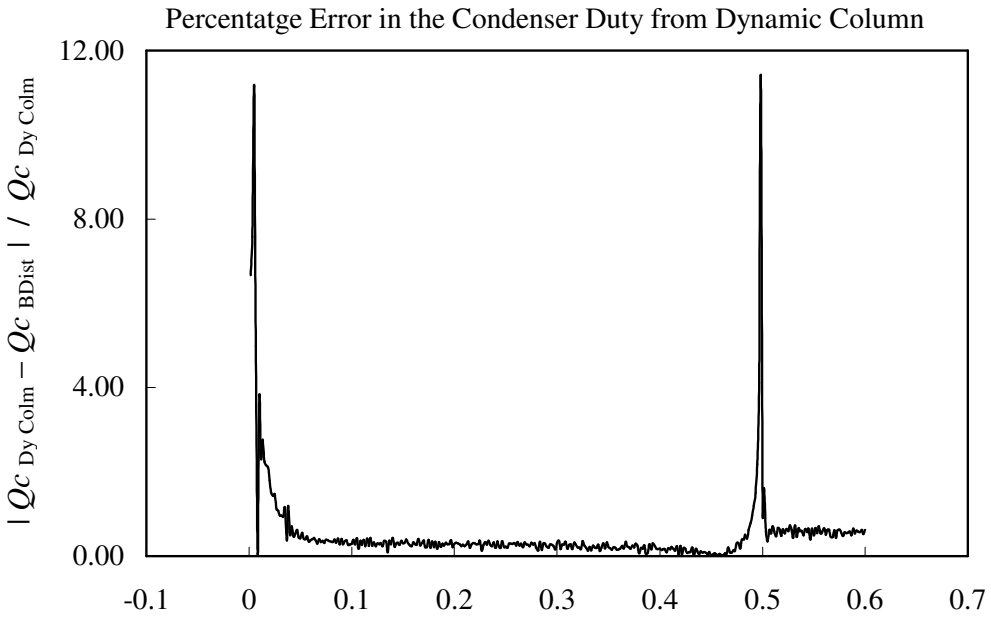


Figure 16.


Percentatge Error in the Condenser Duty from Dynamic Column
| Qc Dy Colm – Qc BDist | / Qc Dy Colm
12.00
8.00
4.00
0.00
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7